NSFOCUS Firewall Series

NF Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

About the NSFOCUS firewall series configuration guides

The NSFOCUS firewall series configuration guides describe the software features for the NSFOCUS firewall series, and guide you through the software configuration procedures. These configuration guides also provide configuration examples to help you apply the software features to different network scenarios.

Applicable hardware and software versions

To obtain software version information for a device, use the **`display version`** command in any view on the device. The configuration guides use the RXX60P28 versions as examples to illustrate feature configuration. For information about feature changes in other software versions, see the release notes for your device.

Contents

Table 1 lists features included in each configuration guide. Support for the features depends on the device model.

Table 1 Configuration guide content

Configuration guide	Content
Fundamentals Configuration Guide	Covers information about using the command line interface, logging in to and setting up the device, and using the basic management features. This guide includes: • CLI (command line interface overview and how to use the CLI) • RBAC • Login management • License management • Device management • FTP and TFTP • File system management • Configuration file management • Software upgrade • ISSU • Automatic • Tcl • Python
Virtual Technologies Configuration Guide	Covers the configuration of virtual technologies. This guide includes: • IRF • Context • Reth interface and redundancy group
Security Configuration Guide	Covers identity authentication, access security, secure management, SSL, and attack protection features. This guide includes: • Security zone • Security policy • ASPF • Session management

Configuration guide	Content
	• Object group • IP source guard • AAA • 802.1X • User identification • Password control • Portal • MAC authentication • IPoE • Public key configuration • PKI • SSH • SSL • Connection limit • Attack detection and prevention • Server connection detection • ARP attack protection • ND attack defense • uRPF • IP-MAC binding • APR • Keychain • Crypto engine • MAC learning through a Layer 3 device • SMS
DPI Configuration Guide	Covers fundamentals and configuration of deep packet inspection. This guide includes: • DPI overview • DPI engine • IPS • URL filtering • Data filtering • File filtering • Anti-virus • Data analysis center • Proxy policy
NAT Configuration Guide	Covers fundamentals and configuration of NAT and AFT. This guide includes: • NAT • NAT66 • AFT
VPN Configuration Guide	Covers VPN-related features. This guide includes: • SSL VPN • IPsec • Tunneling • GRE • L2TP • ADVPN
Internet Access Behavior Management Configuration Guide	Covers Internet access behavior management features. This guide includes:

Configuration guide	Content
	• Bandwidth management • Application audit and management • NetShare control
Load Balancing Configuration Guide	Covers fundamentals, configuration procedures, and configuration examples of load balancing features.
High Availability Configuration Guide	Describes high availability technologies and features available on the device for failure detection and failover.
Interface Configuration Guide	Covers the configuration of various interfaces. This guide includes: • Bulk interface • Ethernet interface • Loopback, null, and inloopback interfaces
Layer 2—LAN Switching Configuration Guide	Covers Layer 2 technologies and features used on a LAN switched network. This guide includes: • MAC address table • Ethernet link aggregation • VLAN • VLAN termination • Spanning tree • LLDP • Layer 2 forwarding
Layer 2—WAN Access Configuration Guide	Covers the Layer 2 WAN access features. This guide includes: • PPP • Mobile communication modem
Layer 3—IP Services Configuration Guide	Covers IP addressing (including static and dynamic IPv4 and IPv6 address assignment), network performance optimization, ARP, and interoperation between IPv4 and IPv6. This guide includes: • IP addressing • IP forwarding basics • Fast forwarding • ARP (including proxy ARP) • IPv6 basics • IPv6 fast forwarding • DHCP • DHCPv6 • DNS • IP performance optimization • Multi-CPU packet distribution • Adjacency table • Web caching
Layer 3—IP Routing Configuration Guide	Covers the routing technologies for IPv4 and IPv6 networks of different sizes, route filtering, route control, and policy based routing. This guide includes: • Basic IP routing • Static routing • IPv6 static routing • RIP • RIPng • OSPF • OSPFv3

Configuration guide	Content
	• IS-IS • BGP • Policy-based routing • IPv6 policy-based routing • Routing policy • Guard route • RIR
ACL and QoS Configuration Guide	Covers information about classifying traffic with ACLs, and controlling traffic and allocating network resources with QoS technologies to improve network performance and network use efficiency. This guide includes: • ACL • QoS (including QoS overview, QoS policy, traffic policing, traffic filtering, and priority marking) • Time range
IP Multicast Configuration Guide	Covers the IP multicast features. This guide includes: • Multicast overview • Multicast routing and forwarding • IGMP • PIM • IPv6 multicast routing and forwarding • MLD
Network Management and Monitoring Configuration Guide	Covers features that help you manage and monitor your network, for example, manage system events, collect traffic statistics, sample packets, assess network performance, and test network connectivity. This guide includes: • Information center • Flow log • Fast log output • NetStream • Cloud connection • Mirroring • Packet capture • NQA • Track • BFD • Monitor Link • Smart Link • Interface backup • Interface collaboration • System maintenance and debugging (ping, tracert, and system debugging) • NTP • EAA • Process monitoring and maintenance • NETCONF • CWMP • SNMP (including the MIB style configuration) • RMON • Event MIB • Process placement

Configuration guide	Content
VPN Instance Configuration Guide	Covers VPN instance fundamentals and configuration procedures.
VXLAN Configuration Guide	Describes the fundamentals and configuration of VXLAN features.
Service Chain Configuration Guide	Describes the fundamentals and configuration of *Service Chain* features.
Acronyms	Lists the significant acronyms in the configuration guides.

NSFOCUS Firewall Series

NF Fundamentals Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes features and tasks that help you get started with the device, iIt includes the following feature modules:
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

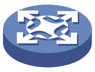
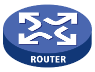
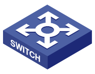
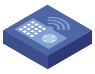
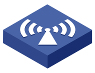
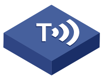
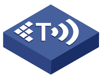
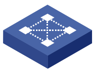
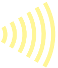
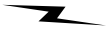
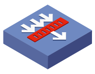
GUI conventions

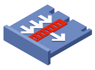
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Using the CLI

About the CLI

At the command-line interface (CLI), you can enter text commands to configure, manage, and monitor the device.

You can use different methods to log in to the CLI. For example, you can log in through the console port or Telnet. For more information about login methods, see "Login overview."

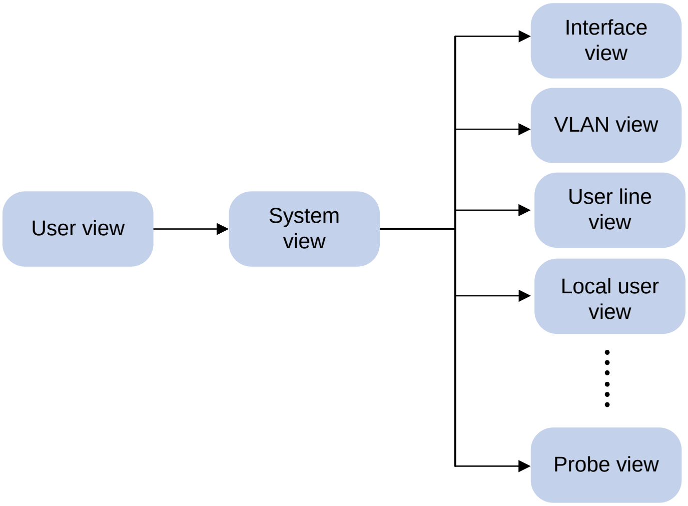
Using CLI views

About CLI views

Commands are grouped in different views by feature. To use a command, you must enter its view.

CLI views are hierarchically organized, as shown in Figure 1. Each view has a unique prompt, from which you can identify where you are and what you can do. For example, the prompt [Sysname-vlan100] shows that you are in VLAN 100 view and can configure attributes for that VLAN.

Figure 1 CLI views



You are placed in user view immediately after you log in to the CLI.

In user view, you can perform the following tasks:

- Perform basic operations including display, debug, file management, FTP, Telnet, clock setting, and reboot.
- Enter system view.

In system view, you can perform the following tasks:

- Configure settings that affect the device as a whole, such as the daylight saving time, banners, and hotkeys.
- Enter feature views.

  For example, you can perform the following tasks:

  - Enter interface view to configure interface parameters.

- Enter VLAN view to add ports to the VLAN.
- Enter user line view to configure login user attributes.

A feature view might have child views. For example, NQA operation view has the child view HTTP operation view.

- Enter probe view by using the **probe** command.

  The probe view provides display, debugging, and maintenance commands, which are mainly used by developers and testers for system fault diagnosis and system operation monitoring.

  For more information about the commands in probe view, see the probe commands manual for each feature.

△ **CAUTION:**

Use the commands in probe view under the guidance of engineers to avoid system exceptions caused by misoperations.

To display all commands available in a view, enter a question mark (?) at the view prompt.

Entering system view from user view

To enter system view from user view, execute the following command:

```
system-view
```

Returning to the upper-level view from a view

Restrictions and guidelines

Executing the **quit** command in user view terminates your connection to the device.

To return from public key view to system view, you must use the **peer-public-key end** command.

Procedure

To return to the upper-level view from a view, execute the following command:

```
quit
```

Returning to user view

About this task

This feature enables you to return to user view from any view except Tcl configuration view and Python shell view by performing a single operation. You do not need to execute the **quit** command multiple times.

To return to user view from Tcl configuration view, use the **tclquit** command. To return to user view from Python shell view, use the **exit()** command.

Procedure

To return directly to user view from any other view except Tcl configuration view and Python shell view, use one of the following methods:

- Execute the **return** command.
- Press **Ctrl+Z**.

Accessing the CLI online help

The CLI online help is context sensitive. Enter a question mark at any prompt or in any position of a command to display all available options.

To access the CLI online help, use one of the following methods:

- Enter a question mark at a view prompt to display the first keyword of every command available in the view. For example:

  ```
  <Sysname> ?
  User view commands:
    archive             Archive configuration
    backup              Backup operation
    boot-loader         Software image file management
  ...
  ```

- Enter a space and a question mark after a command keyword to display all available keywords and arguments.
  - If the question mark is in the place of a keyword, the CLI displays all possible keywords, each with a brief description. For example:

    ```
    <Sysname> terminal ?
      debugging  Enable to display debugging logs on the current terminal
      logging    Display logs on the current terminal
      monitor    Enable to display logs on the current terminal
    ```

  - If the question mark is in the place of an argument, the CLI displays the description for the argument. For example:

    ```
    <Sysname> system-view
    [Sysname] interface vlan-interface ?
      <1-4094>  Vlan-interface interface number
    [Sysname] interface vlan-interface 1 ?
      <cr>
    [Sysname] interface vlan-interface 1
    ```

    <1-4094> is the value range for the argument. **<cr>** indicates that the command is complete and you can press **Enter** to execute the command.

- Enter an incomplete keyword string followed by a question mark to display all keywords starting with that string. The CLI also displays the descriptions for the keywords. For example:

  ```
  <Sysname> f?
    fdisk     Partition a storage medium
    firmware  Firmware management
    fixdisk   Check and repair a storage medium
    format    Format a storage medium
    free      Release a connection
    ftp       Open an FTP connection
    fuser     Display processes that are using a file system, directory, or file
  <Sysname> display ftp?
    ftp         FTP module
    ftp-server  FTP server information
    ftp-user    FTP user information
  ```

Using the undo form of a command

Most configuration commands have an **undo** form for the following tasks:

- Canceling a configuration.
- Restoring the default.
- Disabling a feature.

For example, the **`info-center enable`** command enables the information center. The **`undo info-center enable`** command disables the information center.

Entering a command

When you enter a command, you can perform the following tasks:

- Use keys or hotkeys to edit the command line.
- Use abbreviated keywords or keyword aliases.

Editing a command line

To edit a command line, use the keys listed in Table 1 or the hotkeys listed in Table 4. When you are finished, you can press **Enter** to execute the command.

Table 1 Command line editing keys

Keys	Function
Common keys	If the edit buffer is not full, pressing a common key inserts a character at the cursor and moves the cursor to the right. The edit buffer can store up to 511 characters. Unless the buffer is full, all common characters that you enter before pressing **Enter** are saved in the edit buffer.
Backspace	Deletes the character to the left of the cursor and moves the cursor back one character.
Left arrow key (←)	Moves the cursor one character to the left.
Right arrow key (→)	Moves the cursor one character to the right.
Up arrow key (↑)	Displays the previous command in the command history buffer.
Down arrow key (↓)	Displays the next command in the command history buffer.
Tab	If you press **Tab** after typing part of a keyword, the system automatically completes the keyword. • If a unique match is found, the system displays the complete keyword. • If there is more than one match, press **Tab** multiple times to pick the keyword you want to enter. • If there is no match, the system does not modify what you entered but displays it again in the next line.

The device supports the following special commands:

- **#**–Used by the system in a configuration file as separators for adjacent sections.
- **`version`**–Used by the system in a configuration file to indicate the software version information. For example, **`version 7.1. xxx`**, **`Release xxx`**.

These commands are special because of the following reasons:

- These commands are not intended for you to use at the CLI.

- You can enter the **#** command in any view or the **version** command in system view, or enter any values for them. For example, you can enter **# abc** or **version abc**. However, the settings do not take effect.
- The device does not provide any online help information for these commands.

Entering a text or string type value for an argument

A text type argument value can contain any characters except question marks (?).

A string type argument value can contain any printable characters except question marks (?).

- To include a quotation mark (") or backward slash (\) in a string type argument value, prefix the character with an escape key (\), for example, \" and \\.
- To include a blank space in a string type argument value, enclose the value in quotation marks, for example, "my device".

A specific argument might have more requirements. For more information, see the relevant command reference.

To enter a printable character, you can enter the character or its ASCII code in the range of 32 to 126.

Entering an interface type

You can enter an interface type in one of the following formats:

- Full spelling of the interface type.
- An abbreviation that uniquely identifies the interface type.
- Acronym of the interface type.

For a command line, all interface types are case insensitive. The following table shows the full spellings and acronyms of interface types.

For example, to use the **interface** command to enter the view of interface GigabitEthernet 1/0/1, you can enter the command line in the following formats:

- **interface gigabitethernet 1/0/1**
- **interface g 1/0/1**
- **interface ge 1/0/1**

Spaces between the interface types and interfaces are not required.

Table 2 Full spellings and acronyms of interface types

Full spelling	Acronym
Analogmodem	AM
Async	Asy
IMA-group	IMA-G
Bridge-template	Bridge
Dialer	Dia
Virtual-Access	VA
Inner-Ethernet	I-E
InLoopBack	InLoop
LoopBack	Loop

Full spelling	Acronym
Encrypt	Encry
Ethernet	Eth
GigabitEthernet	GE
VE-Bridge	VEB
Ten-GigabitEthernet	XGE
Virtual-Ethernet	VEth
M-Ethernet	ME
M-GigabitEthernet	MGE
Serial	Ser
subscriber-line	line
Tunnel	Tun
Vlan-interface	Vlan-int
Virtual-Template	VT
InAsync	InAsy
Bridge-Aggregation	BAGG
Register-Tunnel	REG
Route-Aggregation	RAGG
FortyGigE	FGE
HundredGigE	HGE
SAN-Aggregation	SAGG
S-Channel	S-Ch
Virtual-PPP	VPPP
Schannel-Aggregation	SCH-AGG
TwentyGigE	TGE
Twenty-FiveGigE	WGE
Tunnel-Bundle	Tunnel-B
VE-L2VPN	L2VE
VE-L3VPN	L3VE
Blade-Aggregation	BLAGG
Eth-channel	E-Ch
Beth-redundancy	BEth
Reth-redundancy	Reth
Circuit-Emulation	CEM
LoRa-Radio	Lor

Abbreviating commands

You can enter a command line quickly by entering incomplete keywords that uniquely identify the complete command. In user view, for example, commands starting with an **s** include **startup saved-configuration** and **system-view**. To enter the command **system-view**, you need to type only **sy**. To enter the command **startup saved-configuration**, type **st s**.

You can also press **Tab** to complete an incomplete keyword.

Configuring and using command aliases

About this task

You can configure one or more aliases for a command or the starting keywords of commands. Then, you can use the aliases to execute the command or commands. If the command or commands have **undo** forms, you can also use the aliases to execute the **undo** command or commands.

For example, if you configure the **shiprt** alias for **display ip routing-table**, you can enter **shiprt** to execute the **display ip routing-table** command. If you configure the **ship** alias for **display ip**, you can use **ship** to execute all commands starting with **display ip**, including:

- Enter **ship routing-table** to execute the **display ip routing-table** command.
- Enter **ship interface** to execute the **display ip interface** command.

The device provides a set of system-defined command aliases, as listed in Table 3.

Table 3 System-defined command aliases

Command alias	Command or command keyword
access-list	**acl**
end	**return**
erase	**delete**
exit	**quit**
hostname	**sysname**
logging	**info-center**
no	**undo**
show	**display**
write	**save**

Restrictions and guidelines

A command alias can be used only as the first keyword of a command or the second keyword of the **undo** form of a command.

After you successfully execute a command by using an alias, the system saves the command, instead of the alias, to the running configuration.

The command string can include up to nine parameters. Each parameter starts with the dollar sign ($) and a sequence number in the range of 1 to 9. For example, you can configure the alias **shinc** for the **display ip $1 | include $2** command. Then, to execute the **display ip routing-table | include Static** command, you need to enter only **shinc routing-table Static**.

To use an alias for a command that has parameters, you must specify a value for each parameter. If you fail to do so, the system informs you that the command is incomplete and displays the command string represented by the alias.

System-defined command aliases cannot be deleted.

Procedure

1. Enter system view.

   `system-view`

2. Configure a command alias.

   `alias` *`alias command`*

   By default, the device has a set of command aliases, as listed in Table 3.

3. (Optional.) Display command aliases.

   `display alias` `[` *`alias`* `]`

   This command is available in any view.

Configuring and using hotkeys

About this task

The system defines the hotkeys shown in Table 4 and provides a set of configurable command hotkeys. Pressing a command hotkey is the same as entering a command.

Table 4 System-reserved hotkeys

Hotkey	Function
Ctrl+A	Moves the cursor to the beginning of a line.
Ctrl+B	Moves the cursor one character to the left.
Ctrl+C	Stops the current command.
Ctrl+D	Deletes the character at the cursor.
Ctrl+E	Moves the cursor to the end of a line.
Ctrl+F	Moves the cursor one character to the right.
Ctrl+H	Deletes the character to the left of the cursor.
Ctrl+K	Aborts the connection request.
Ctrl+N	Displays the next command in the history buffer.
Ctrl+P	Displays the previous command in the history buffer.
Ctrl+R	Redisplays the current line.
Ctrl+V	Pastes text from the clipboard.
Ctrl+W	Deletes the word to the left of the cursor.
Ctrl+X	Deletes all characters to the left of the cursor.
Ctrl+Y	Deletes all characters from the cursor to the end of the line.
Ctrl+Z	Returns to user view.
Ctrl+]	Terminates the current connection.
Esc+B	Moves the cursor back one word.
Esc+D	Deletes all characters from the cursor to the end of the word.

Hotkey	Function
Esc+F	Moves the cursor forward one word.
Esc+N	Moves the cursor down one line. You can use this hotkey before pressing **Enter**.
Esc+P	Moves the cursor up one line. You can use this hotkey before pressing **Enter**.
Esc+<	Moves the cursor to the beginning of the clipboard.
Esc+>	Moves the cursor to the end of the clipboard.

Restrictions and guidelines

A hotkey can correspond to only one command or function. If you assign multiple commands or functions to the same hotkey, the most recently assigned command or function takes effect.

A command or function can be assigned to multiple hotkeys. You can use any of the hotkeys to execute the command or function.

If a hotkey is also defined by the terminal software you are using to interact with the device, the terminal software definition takes effect.

Procedure

1. Enter system view.

   `system-view`

2. Assign a command to a configurable command hotkey.

   `hotkey` { **`ctrl_g`** | **`ctrl_l`** | **`ctrl_o`** | **`ctrl_t`** | **`ctrl_u`** } *command*

   The following are the defaults:

   - **Ctrl+G** is assigned the **`display current-configuration`** command.
   - **Ctrl+L** is assigned the **`display ip routing-table`** command.
   - **Ctrl+O** is assigned the **`undo debugging all`** command.
   - No command is assigned to **Ctrl+T** or **Ctrl+U**.

3. (Optional.) Display hotkeys.

   `display hotkey`

   This command is available in any view.

Enabling redisplaying entered-but-not-submitted commands

About this task

Your input might be interrupted by system information output. If redisplaying entered-but-not-submitted commands is enabled, the system redisplays your input after finishing the output. You can then continue entering the command line.

Procedure

1. Enter system view.

   `system-view`

2. Enable redisplaying entered-but-not-submitted commands.

   `info-center synchronous`

   By default, the system does not redisplay entered-but-not-submitted commands.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

Understanding command-line syntax error messages

After you press **Enter** to submit a command, the command line interpreter examines the command syntax.

- If the command passes syntax check, the CLI executes the command.
- If the command fails syntax check, the CLI displays an error message.

Table 5 Common command-line syntax error messages

Error message	Cause
% Unrecognized command found at '^' position.	The keyword in the marked position is invalid.
% Incomplete command found at '^' position.	One or more required keywords or arguments are missing.
% Ambiguous command found at '^' position.	The entered character sequence matches more than one command.
% Too many parameters found at '^' position.	The entered character sequence contains excessive keywords or arguments.
% Wrong parameter found at '^' position.	The argument in the marked position is invalid.

Using the command history feature

About command history buffers

The system automatically saves commands successfully executed by a login user to the following two command history buffers:

- Command history buffer for the user line.
- Command history buffer for all user lines.

Table 6 Comparison between the two types of command history buffers

Item	Command history buffer for a user line	Command history buffer for all user lines
Which commands are saved in the buffer?	Commands successfully executed by the current user of the user line.	Commands successfully executed by all login users.
Can commands in the buffer be displayed?	Yes.	Yes.
Can commands in the buffer be recalled?	Yes.	No.
Are buffered commands cleared when the user logs out?	Yes.	No.
Is the buffer size adjustable?	Yes.	No. The buffer size is fixed at 1024.

Command buffering rules

The system follows these rules when buffering commands:

- If you use incomplete keywords when entering a command, the system buffers the command in the exact form that you used.
- If you use an alias when entering a command, the system transforms the alias to the represented command or command keywords before buffering the command.
- If you enter a command in the same format multiple times in succession, the system buffers the command only once. If you enter a command in different formats multiple times, the system buffers each command format. For example, **`display cu`** and **`display current-configuration`** are buffered as two entries but successive repetitions of **`display cu`** create only one entry.
- To buffer a new command when a buffer is full, the system deletes the oldest command entry in the buffer.

Managing and using the command history buffers

Displaying the commands in command history buffers

To display the commands in command history buffers, execute the following commands in any view:

- Display the commands in command history buffers for a user line.

  ```
  display history-command
  ```

- Display the commands in command history buffers for all user lines.

  ```
  display history-command all
  ```

Recalling commands in the command history buffer for a user line

Use up and down arrow keys to navigate to the command and press **Enter**.

Setting the size of the command history buffer for a user line

Use the **`history-command max-size`** command in user line or user line class view. For more information, see *Fundamentals Command Reference*.

Repeating commands in the command history buffer for a user line

About this task

You can recall and execute commands in the command history buffer for the current user line multiple times.

Restrictions and guidelines

The **`repeat`** command is available in any view. However, to repeat a command, you must first enter the view for the command. To repeat multiple commands, you must first enter the view for the first command.

The **`repeat`** command executes commands in the order they were executed.

The system waits for your interaction when it repeats an interactive command.

Procedure

To repeat commands in the command history buffer for the current user line, execute the following command:

```
repeat [ number ] [ count times ] [ delay seconds ]
```

Controlling the CLI output

This section describes the CLI output control features that help you identify the desired output.

Pausing between screens of output

About this task

The device can automatically pause after displaying a specific number of lines if the output is too long to fit on one screen. At a pause, the device displays **----more----**. You can use the keys described in Table 7 to display more information or stop the display.

You can also disable pausing between screens of output for the current session. Then, all output is displayed at one time and the screen is refreshed continuously until the final screen is displayed.

Table 7 Output controlling keys

Keys	Function
Space	Displays the next screen.
Enter	Displays the next line.
Ctrl+C	Stops the display and cancels the command execution.
<PageUp>	Displays the previous page.
<PageDown>	Displays the next page.

Disabling pausing between screens of output

To disable pausing between screens of output, execute the following command in user view:

```
screen-length disable
```

The default depends on the settings of the **screen-length** command in user line view. The following are the default settings for the **screen-length** command:

- Pausing between screens of output is enabled.
- The maximum number of lines to be displayed at a time is 24.

For more information about the **screen-length** command, see *Fundamentals Command Reference*.

This command is a one-time command and takes effect only for the current CLI session.

Numbering each output line from a display command

About this task

For easy identification, you can use the **| by-linenum** option to display a number for each output line from a **display** command.

Each line number is displayed as a 5-character string and might be followed by a colon (:) or hyphen (-). If you specify both **| by-linenum** and **| begin** *regular-expression* for a **display** command, a hyphen is displayed for all lines that do not match the regular expression.

Procedure

To number each output line from a **display** command, execute the following command in any view:

display *command* **| by-linenum**

Example

# Display information about VLAN 999, numbering each output line.

```
<Sysname> display vlan 999 | by-linenum
    1:  VLAN ID: 999
    2:  VLAN type: Static
    3:  Route interface: Configured
    4:  IPv4 address: 192.168.2.1
    5:  IPv4 subnet mask: 255.255.255.0
    6:  Description: For LAN Access
    7:  Name: VLAN 0999
    8:  Tagged ports:   None
    9:  Untagged ports: None
```

Filtering the output from a display command

About this task

You can use the **|** { **begin** | **exclude** | **include** } *regular-expression* option to filter the output from a **display** command.

- **begin**—Displays the first line matching the specified regular expression and all subsequent lines.
- **exclude**—Displays all lines not matching the specified regular expression.
- **include**—Displays all lines matching the specified regular expression.
- *regular-expression*—A case-sensitive string of 1 to 256 characters, which can contain the special characters described in Table 8.

Table 8 Special characters supported in a regular expression

Characters	Meaning	Examples
^	Matches the beginning of a line.	"^u" matches all lines beginning with "u". A line beginning with "Au" is not matched.
$	Matches the end of a line.	"u$" matches all lines ending with "u". A line ending with "uA" is not matched.
. (period)	Matches any single character.	".s" matches "as" and "bs".
*	Matches the preceding character or string zero, one, or multiple times.	"zo*" matches "z" and "zoo", and "(zo)*" matches "zo" and "zozo".
+	Matches the preceding character or string one or multiple times.	"zo+" matches "zo" and "zoo", but not "z".
|	Matches the preceding or succeeding string.	"def|int" matches a line containing "def" or "int".
( )	Matches the string in the parentheses, usually used together with the plus sign (+) or asterisk sign (*).	"(123A)" matches "123A". "408(12)+" matches "40812" and "408121212", but not "408".
\N	Matches the preceding strings in parentheses, with the *N*th string repeated once.	"(string)\1" matches a string containing "stringstring". "(string1)(string2)\2" matches a string containing

Characters	Meaning	Examples
		"string1string2string2". "(string1)(string2)\1\2" matches a string containing " string1string2string1string2".
[ ]	Matches a single character in the brackets.	"[16A]" matches a string containing 1, 6, or A; "[1-36A]" matches a string containing 1, 2, 3, 6, or A (- is a hyphen). To match the character "]", put it immediately after "[", for example, [ ]abc]. There is no such limit on "[".
[^]	Matches a single character that is not in the brackets.	"[^16A]" matches a string that contains one or more characters except for 1, 6, or A, such as "abc". A match can also contain 1, 6, or A (such as "m16"), but it cannot contain these three characters only (such as 1, 16, or 16A).
{n}	Matches the preceding character *n* times. The number *n* must be a nonnegative integer.	"o{2}" matches "food", but not "Bob".
{n,}	Matches the preceding character *n* times or more. The number *n* must be a nonnegative integer.	"o{2,}" matches "foooood", but not "Bob".
{n,m}	Matches the preceding character *n* to *m* times or more. The numbers *n* and *m* must be nonnegative integers and *n* cannot be greater than *m*.	" o{1,3}" matches "fod", "food", and "foooood", but not "fd".
\<	Matches a string that starts with the pattern following \<. A string that contains the pattern is also a match if the characters preceding the pattern are not digits, letters, or underscores.	"\<do" matches "domain" and "doa".
\>	Matches a string that ends with the pattern preceding \>. A string that contains the pattern is also a match if the characters following the pattern are not digits, letters, or underscores.	"do\>" matches "undo" and "cdo".
\b	Matches a word that starts with the pattern following \b or ends with the pattern preceding \b.	"er\b" matches "never", but not "verb" or "erase". "\ber" matches "erase", but not "verb" or "never".
\B	Matches a word that contains the pattern but does not start or end with the pattern.	"er\B" matches "verb", but not "never" or "erase".
\w	Same as [A-Za-z0-9_], matches a digit, letter, or underscore.	"v\w" matches "vlan" and "service".
\W	Same as [^A-Za-z0-9_], matches a character that is not a digit, letter, or underscore.	"\Wa" matches "-a", but not "2a" or "ba".
\	Escape character. If a special character listed in this table follows \, the specific meaning of the character is removed.	"\\" matches a string containing "\", "\^" matches a string containing "^", and "\\b" matches a string containing "\b".

Restrictions and guidelines

The required filtering time increases with the complexity of the regular expression. To abort the filtering process, press **Ctrl+C**.

Examples

# Display the running configuration, starting from the first configuration line that contains **line**.

```
<Sysname> display current-configuration | begin line
line class console
user-role network-admin
#
line class vty
user-role network-operator
#
line console 0
user-role network-admin
#
line vty 0 31
authentication-mode scheme
user-role network-operator
#
...
```

# Display brief information about interfaces in up state.

```
<Sysname> display interface brief | exclude DOWN
Brief information on interfaces in route mode:
Link: ADM - administratively down; Stby - standby
Protocol: (s) - spoofing
Interface            Link Protocol Primary IP      Description
InLoop0              UP   UP(s)    --
NULL0                UP   UP(s)    --
Vlan1                UP   UP       192.168.1.83
```

# Display SNMP-related running configuration lines.

```
<Sysname> display current-configuration | include snmp
snmp-agent
 snmp-agent community write private
 snmp-agent community read public
 snmp-agent sys-info version all
 snmp-agent target-host trap address udp-domain 192.168.1.26 params securityname public
```

Saving the output from a display command to a file

About this task

A **display** command shows certain configuration and operation information of the device. Its output might vary over time or with user configuration or operation. You can save the output to a file for future retrieval or troubleshooting.

Use one of the following methods to save the output from a **display** command:

- Save the output to a separate file. Use this method if you want to use one file for a single **display** command.

- Append the output to the end of a file. Use this method if you want to use one file for multiple **display** commands.

Procedure

To save the output from a **display** command to a file, use one of the following commands in any view:

- Save the output from a **display** command to a separate file.

  display *command* **>** *filename*

- Append the output from a **display** command to the end of a file.

  display *command* **>>** *filename*

Examples

# Save the VLAN 1 settings to a separate file named **vlan.txt**.

```
<Sysname> display vlan 1 > vlan.txt
```

# Verify that the VLAN 1 settings are saved to the file **vlan.txt**.

```
<Sysname> more vlan.txt
VLAN ID: 1
 VLAN type: Static
 Route interface: Not configured
 Description: VLAN 0001
 Name: VLAN 0001
 Tagged ports:   None
 Untagged ports: None
```

# Append the VLAN 999 settings to the end of the file **vlan.txt**.

```
<Sysname> display vlan 999 >> vlan.txt
```

# Verify that the VLAN 999 settings are appended to the end of the file **vlan.txt**.

```
<Sysname> more vlan.txt
VLAN ID: 1
 VLAN type: Static
 Route interface: Not configured
 Description: VLAN 0001
 Name: VLAN 0001
 Tagged ports:   None
 Untagged ports: None

 VLAN ID: 999
 VLAN type: Static
 Route interface: Configured
 IPv4 address: 192.168.2.1
 IPv4 subnet mask: 255.255.255.0
 Description: For LAN Access
 Name: VLAN 0999
 Tagged ports:   None
 Untagged ports: None
```

Viewing and managing the output from a display command effectively

You can use the following methods in combination to filter and manage the output from a **display** command:

- Numbering each output line from a display command
- Filtering the output from a display command
- Saving the output from a display command to a file

Procedure

To use multiple measures to view and manage the output from a **display** command effectively, execute the following command in any view:

display *command* [**|** [**by-linenum**] { **begin** | **exclude** | **include** } *regular-expression*] [**>** *filename* | **>>** *filename*]

Examples

# Save the running configuration to a separate file named **test.txt**, with each line numbered.

```
<Sysname> display current-configuration | by-linenum > test.txt
```

# Append lines including **snmp** in the running configuration to the file **test.txt**.

```
<Sysname> display current-configuration | include snmp >> test.txt
```

# Display the first line that begins with **user-group** in the running configuration and all the following lines.

```
<Sysname> display current-configuration | by-linenum begin user-group
  114:  user-group system
  115-  #
  116-  return
```

// The colon (:) following a line number indicates that the line contains the string user-group. The hyphen (-) following a line number indicates that the line does not contain the string **user-group**.

Contents

Configuring RBAC

About RBAC

Role-based access control (RBAC) controls access permissions of users based on user roles.

RBAC assigns access permissions to user roles that are created for different job functions. Users are given permission to access a set of items and resources based on the users' user roles. Separating permissions from users enables simple permission authorization management.

Permission assignment

Use the following methods to assign permissions to a user role:

- Define a set of rules to determine accessible or inaccessible items for the user role. (See "User role rules.")
- Configure resource access policies to specify which resources are accessible to the user role. (See "Resource access policies.")

To use a command related to a system resource, a user role must have access to both the command and the resource.

For example, a user role has access to the **vlan** command and access only to VLAN 10. When the user role is assigned, you can use the **vlan** command to create VLAN 10 and enter its view. However, you cannot create any other VLANs. If the user role has access to VLAN 10 but does not have access to the **vlan** command, you cannot use the command to enter the view of VLAN 10.

When a user logs in to the device with any user role and enters **<?>** in a view, help information is displayed for the system-defined command aliases in the view. However, the user might not have the permission to access the command aliases. Whether the user can access the command aliases depends on the user role's permission to the commands corresponding to the aliases. For information about command aliases, see "Using the CLI."

A user that logs in to the device with any user role has access to the **system-view**, **quit**, and **exit** commands.

User role rules

User role rules permit or deny access to the items, including commands, Web pages, XML elements, or MIB nodes. You can define the following types of rules for different access control granularities:

- **Command rule**—Controls access to a command or a set of commands that match a regular expression.
- **Feature rule**—Controls access to the commands of a feature by command type.
- **Feature group rule**—Controls access to the commands of features in a feature group by command type.
- **Web menu rule**—Controls access to Web pages used for configuring the device. These Web pages are called Web menus.
- **XML element rule**—Controls access to XML elements used for configuring the device.
- **OID rule**—Controls SNMP access to a MIB node and its child nodes. An OID is a dotted numeric string that uniquely identifies the path from the root node to a leaf node.

The items (commands, Web menus, XML elements, and MIB nodes) are controlled based on the following types:

- **Read**—Items that display configuration and maintenance information. For example, the **display** commands and the **dir** command.

- **Write**—Items that configure the features in the system. For example, the **info-center enable** command and the **debugging** command.
- **Execute**—Items that execute specific functions. For example, the **ping** command and the **ftp** command.

A user role can access the set of permitted items specified in the user role rules. The user role rules include predefined (identified by sys-*n*) and user-defined user role rules. For more information about the user role rule priority, see "Configuring user role rules."

Resource access policies

Resource access policies control access of a user role to system resources and include the following types:

- **Interface policy**—Controls access to interfaces.
- **VLAN policy**—Controls access to VLANs.
- **VPN instance policy**—Controls access to VPN instances.
- **Security zone policy**—Controls access to security zones.

Resource access policies do not control access to the VLAN, VPN instance, security zone, or interface options in the **display** commands. You can specify these options in the **display** commands if the options are permitted by any user role rule.

Predefined user roles

The system provides predefined user roles. These user roles have access to all system resources. However, their access permissions differ, as shown in Table 1.

Among all of the predefined user roles, only network-admin, context-admin, and level-15 can create, modify, and delete local users and local user groups. The other user roles can only modify their own passwords if they have permissions to configure local users and local user groups.

All the predefined user roles are available for the default context. The network-admin and network-operator user roles are not available for non-default contexts. For more information about contexts, see *Virtual Technologies Configuration Guide*.

The access permissions of the level-0 to level-14 user roles can be modified through user role rules and resource access policies. However, you cannot make changes on the predefined access permissions of these user roles. For example, you cannot change the access permission of these user roles to the **display history-command all** command.

Table 1 Predefined roles and permissions matrix

User role name	Permissions
network-admin	Accesses all features and resources in the system, except for the **display security-logfile summary**, **info-center security-logfile directory**, and **security-logfile save** commands.
network-operator	• Accesses the **display** commands for features and resources in the system. To display all accessible commands of the user role, use the **display role** command. • Enables local authentication login users to change their own passwords. • Accesses the command used for entering XML view. • Accesses all read-type Web menu items. • Accesses all read-type XML elements. • Accesses all read-type MIB nodes.
level-*n* (*n* = 0 to 15)	• **level-0**—Has access to commands including **ping**, **tracert**, **ssh2**, **telnet**, and **super**. Level-0 access rights are

User role name	Permissions
	configurable. • **level-1**—Has access to the **`display`** commands of features and resources in the system. The **level-1** user role also has all access rights of the level-0 user role. Level-1 access rights are configurable. • **level-2 to level-8, and level-10 to level-14**—Have no access rights by default. Access rights are configurable. • **level-9**—Has access to most of the features and resources in the system. If you are logged in with a local user account that has a level-9 user role, you can change the password in the local user account. The following are the major features and commands that the level-9 user role cannot access: ○ RBAC non-debugging commands. ○ Local users. ○ File management. ○ Device management. ○ The **`display history-command all`** command. • **level-15**—Has the same rights as network-admin on the default context, and has the same rights as context-admin on non-default contexts.
security-audit	Security log manager. The user role has the following access rights to security log files: • Accesses the commands for displaying and maintaining security log files (for example, the **`dir`**, **`display security-logfile summary`**, and **`more`** commands). • Accesses the commands for managing security log files and security log file system (for example, the **`info-center security-logfile directory`**, **`mkdir`**, and **`security-logfile save`** commands). For more information about security log management commands, see information center commands in *Network Management and Monitoring Command Reference*. For more information about file system management commands, see *Fundamentals Command Reference*. **IMPORTANT:** Only the security-audit user role has access to security log files. You cannot assign the security-audit user role to non-AAA authentication users.
guest-manager	Accesses only guest-related web pages, and has no access to commands.
context-admin	Accesses all features and resources in the context, except for the **`display security-logfile summary`**, **`info-center security-logfile directory`**, and **`security-logfile save`** commands.
context-operator	• Accesses the **`display`** commands for features and resources available in the context. To display all accessible commands of the user role, use the **`display role`** command. • Enables local authentication login users to change their own passwords. • Accesses the command used for entering XML view. • Accesses all read-type Web menu items. • Accesses all read-type XML elements. • Accesses all read-type MIB nodes.
system-admin	System manager. The user role has the following access permissions:

User role name	Permissions
	• Has read, write, and execute access permissions to the Web menus under the Summary category. • Has read, write, and execute access permissions to the System Logs Web menu under the Monitoring category. • Has the following access permissions under the System category: ○ Has read access permissions to the Administrators and Roles Web menus. ○ Has read, write, and execute access permissions to other Web menus. • Accesses the **`ping`** and **`tracert`** commands. To display the detailed access permissions of this user role, use the **`display role`** command. **IMPORTANT:** You cannot assign the system-admin user role to non-AAA authentication users.
security-admin	Security manager. The user role has the following access permissions: • Has read, write, and execute access permissions to Web menus under the Policies, Objects, and Network categories. • Has the following access permissions under the Monitoring category: ○ Has no access permissions to the System Logs and Operation Logs Web menus. ○ Has read, write, and execute access permissions to other Web menus. • Accesses the **`ping`** and **`tracert`** commands. To display the detailed access permissions of this user role, use the **`display role`** command. **IMPORTANT:** You cannot assign the security-admin user role to non-AAA authentication users.
audit-admin	Audit manager. The user role has the following access permissions: • Has read, write, and execute access permissions to the Operation Logs Web menu under the Monitoring category. • Accesses the **`ping`** and **`tracert`** commands. To display the detailed access permissions of this user role, use the **`display role`** command. **IMPORTANT:** You cannot assign the audit-admin user role to non-AAA authentication users.

User role assignment

You assign access rights to a user by assigning a minimum of one user role. The user can use the collection of items and resources accessible to all user roles assigned to the user. For example, you can access any interface to use the **`qos apply policy`** command if you are assigned the following user roles:

- User role A denies access to the **`qos apply policy`** command and permits access only to interface GigabitEthernet 1/0/1.
- User role B permits access to the **`qos apply policy`** command and all interfaces.

Depending on the authentication method, user role assignment has the following methods:

- **AAA authorization**—If scheme authentication is used, the AAA module handles user role assignment.
  - If the user passes local authorization, the device assigns the user roles specified in the local user account.
  - If the user passes remote authorization, the remote AAA server assigns the user roles specified on the server. The AAA server can be a RADIUS or HWTACACS server.
- **Non-AAA authorization**—When the user accesses the device without authentication or by passing password authentication on a user line, the device assigns user roles specified on the user line. This method also applies to SSH clients that use publickey or password-publickey authentication. User roles assigned to these SSH clients are specified in their respective device management user accounts.

For more information about AAA, see *Security Configuration Guide*. For more information about user lines, see "Login overview" and "Configuring CLI login."

RBAC tasks at a glance

To configure RBAC, perform the following tasks:

1. Creating a user role
2. Configuring user role rules
3. (Optional.) Configuring a feature group
4. Configuring resource access policies
   - Configuring the user role interface policy
   - Configuring the user role VLAN policy
   - Configuring the user role VPN instance policy
   - Configuring the user role security zone policy
5. Assigning user roles
   - Enabling the default user role feature
   - Assigning user roles to remote AAA authentication users
   - Assigning user roles to local AAA authentication users
   - Assigning user roles to non-AAA authentication users on user lines
6. Configuring temporary user role authorization
   a. Setting the authentication mode for temporary user role authorization
   b. Specifying the default target user role for temporary user role authorization
   c. Setting an authentication password for temporary user role authorization
   d. (Optional.) Automatically obtaining the login username for temporary user role authorization
   e. Obtaining temporary user role authorization

Creating a user role

About this task

In addition to the predefined user roles, you can create a maximum of 64 custom user roles for granular access control.

Procedure

1. Enter system view.

```
system-view
```

2. Create a user role and enter its view.

   role name *role-name*

   By default, the system has the following predefined user roles:

   - network-admin.
   - network-operator.
   - context-admin.
   - context-operator.
   - level-*n* (where *n* equals an integer in the range of 0 to 15).
   - security-audit.
   - guest-manager.
   - system-admin.
   - security-admin.
   - audit-admin.

   Among these user roles, only the permissions and descriptions of the level-0 to level-14 user roles are configurable.

3. (Optional.) Configure a description for the user role.

   description *text*

   By default, a user role does not have a description.

Configuring user role rules

About this task

You can configure user role rules to permit or deny the access of a user role to specific commands, Web pages, XML elements, and MIB nodes.

The following guidelines apply to non-OID rules:

- If two user-defined rules of the same type conflict, the rule with the higher ID takes effect. For example, a user role can use the **tracert** command but not the **ping** command if the user role contains rules configured by using the following commands:
  - **rule** *1* **permit command** *ping*
  - **rule** *2* **permit command** *tracert*
  - **rule** *3* **deny command** *ping*
- If a predefined user role rule and a user-defined user role rule conflict, the user-defined user role rule takes effect.

The following guidelines apply to OID rules:

- The system compares an OID with the OIDs specified in user role rules, and it uses the longest match principle to select a rule for the OID. For example, a user role cannot access the MIB node with OID 1.3.6.1.4.1.25506.141.3.0.1 if the user role contains rules configured by using the following commands:
  - **rule** *1* **permit read write oid** *1.3.6*
  - **rule** *2* **deny read write oid** *1.3.6.1.4.1*
  - **rule** *3* **permit read write oid** *1.3.6.1.4*
- If the same OID is specified in multiple rules, the rule with the higher ID takes effect. For example, a user role can access the MIB node with OID 1.3.6.1.4.1.25506.141.3.0.1 if the user role contains rules configured by using the following commands:
  - **rule** *1* **permit read write oid** *1.3.6*
  - **rule** *2* **deny read write oid** *1.3.6.1.4.1*

  - **rule** *3* **permit read write oid** *1.3.6.1.4.1*

Restrictions and guidelines

- Context-related rules take effect only if they are configured in views of the following user roles: network-admin, network-operator, context-admin, context-operator, and level-15.
- Only the network-admin, context-admin, and level-15 user roles have access to the following commands:
  - The **display history-command all** command.
  - All commands that start with the **display role**, **display license**, **reboot**, **startup saved-configuration**, and **undo startup saved-configuration** keywords.
  - All commands that start with the **role**, **undo role**, **super**, **undo super**, **license**, **password-recovery**, and **undo password-recovery** keywords in system view.
  - All commands that start with the **snmp-agent community**, **undo snmp-agent community**, **snmp-agent usm-user**, **undo snmp-agent usm-user**, **snmp-agent group**, and **undo snmp-agent group** keywords in system view.
  - All commands that start with the **user-role**, **undo user-role**, **authentication-mode**, **undo authentication-mode**, **set authentication password**, and **undo set authentication password** keywords in user line view or user line class view.
  - All commands that start with the **user-role** and **undo user-role** keywords in schedule view or in CLI-defined policy view.
  - All commands of the event MIB feature.
- You can configure a maximum of 256 user-defined rules for a user role. The total number of user-defined user role rules cannot exceed 1024.
- Any rule modification, addition, or removal for a user role takes effect only on users who are logged in with the user role after the change.

Procedure

1. Enter system view.

   system-view

2. Enter user role view.

   role name *role-name*

3. Configure rules for the user role. Choose the options to configure as needed:
   - Configure a command rule.

     rule *number* { **deny** | **permit** } **command** *command-string*
   - Configure a feature rule.

     rule *number* { **deny** | **permit** } { **execute** | **read** | **write** } * **feature** [*feature-name*]
   - Configure a feature group rule.

     rule *number* { **deny** | **permit** } { **execute** | **read** | **write** } * **feature-group** *feature-group-name*

     A feature group rule takes effect only after the feature group is created.
   - Configure a Web menu rule.

     rule *number* { **deny** | **permit** } { **execute** | **read** | **write** } * **web-menu** [*web-string*]
   - Configure an XML element rule.

rule *number* { **deny** | **permit** } { **execute** | **read** | **write** } * **xml-element** [*xml-string*]

- Configure an OID rule.

  rule *number* { **deny** | **permit** } { **execute** | **read** | **write** } * **oid** *oid-string*

Configuring a feature group

About this task

Use feature groups to bulk assign command access permissions to sets of features. In addition to the predefined feature groups, you can create a maximum of 64 custom feature groups and assign a feature to multiple feature groups.

Procedure

1. Enter system view.

   system-view

2. Create a feature group and enter its view.

   role feature-group name *feature-group-name*

   By default, the system has the following predefined feature groups, which cannot be deleted or modified:

   - **L2**—Includes all Layer 2 commands.
   - **L3**—Includes all Layer 3 commands.

3. Add a feature to the feature group.

   feature *feature-name*

   By default, a feature group does not have any feature.

Configuring resource access policies

About resource access policies

Every user role has one VLAN policy, VPN instance policy, security zone policy, and interface policy. By default, these policies permit a user role to access any system resources. You can configure the policies of a user-defined user role or a predefined level-*n* user role to limit its access to any resources.

Restrictions and guidelines for resource access policy configuration

The policy configuration takes effect only on users who are logged in with the user role after the configuration.

Configuring the user role interface policy

1. Enter system view.

   system-view

2. Enter user role view.

   role name *role-name*

3. Enter user role interface policy view.

```
interface policy deny
```

By default, the interface policy of the user role permits access to all interfaces.

△ **CAUTION:**

This command denies the user role access to any interfaces if you do not specify accessible interfaces by using the **`permit interface`** command.

4. (Optional.) Specify a list of interfaces accessible to the user role.

   `permit interface` *`interface-list`*

   By default, no accessible interfaces are configured in user role interface policy view.

   Repeat this step to add multiple accessible interfaces.

Configuring the user role VLAN policy

1. Enter system view.

   `system-view`

2. Enter user role view.

   `role name` *`role-name`*

3. Enter user role VLAN policy view.

   `vlan policy deny`

   By default, the VLAN policy of the user role permits access to all VLANs.

△ **CAUTION:**

This command denies the user role access to any VLANs if you do not specify accessible VLANs by using the **`permit vlan`** command.

4. (Optional.) Specify a list of VLANs accessible to the user role.

   `permit vlan` *`vlan-id-list`*

   By default, no accessible VLANs are configured in user role VLAN policy view.

   Repeat this step to add multiple accessible VLANs.

Configuring the user role VPN instance policy

1. Enter system view.

   `system-view`

2. Enter user role view.

   `role name` *`role-name`*

3. Enter user role VPN instance policy view.

   `vpn-instance policy deny`

   By default, the VPN instance policy of the user role permits access to all VPN instances.

△ **CAUTION:**

This command denies the user role access to any VPN instances if you do not specify accessible VPN instances by using the **`permit vpn-instance`** command.

4. (Optional.) Specify a list of VPN instances accessible to the user role.

   `permit vpn-instance` *`vpn-instance-name`*`&<1-10>`

By default, no accessible VPN instances are configured in user role VPN instance policy view.

Repeat this step to add multiple accessible VPN instances.

Configuring the user role security zone policy

1. Enter system view.

   system-view

2. Enter user role view.

   role name *role-name*

3. Enter user role security zone policy view.

   security-zone policy deny

   By default, the security zone policy of the user role permits access to all security zones.

> △ **CAUTION:**
>
> This command denies the user role access to any security zones if you do not specify accessible security zones by using the **permit security-zone** command.

4. (Optional.) Specify a list of security zones accessible to the user role.

   permit security-zone *security-zone-name*&<1-10>

   By default, no accessible security zones are configured in user role security zone policy view.

   Repeat this step to add multiple accessible security zones.

Assigning user roles

Restrictions and guidelines for user role assignment

To control user access to the system, you must assign a minimum of one user role. Make sure a minimum of one user role among the user roles assigned by the server exists on the device.

Enabling the default user role feature

About this task

The default user role feature assigns the default user role to AAA-authenticated users if the authentication server (local or remote) does not assign any user roles to the users. These users are allowed to access the system with the default user role.

You can specify any user role existing in the system as the default user role.

Procedure

1. Enter system view.

   system-view

2. Enable the default user role feature.

   role default-role enable [*role-name*]

   By default, the default user role feature is disabled.

   If you do not use the **authorization-attribute** command to assign user roles to local users, you must enable the default user role feature. For information about the **authorization-attribute** command, see AAA commands in *Security Command Reference*.

Assigning user roles to remote AAA authentication users

For remote AAA authentication users, user roles are specified on the remote authentication server. For information about specifying user roles for RADIUS users on the RADIUS server, see the RADIUS server documentation. For HWTACACS users, the user roles must be specified in the **roles="***role-1 role-2 … role-n***"** format, where user roles are space separated. For example, specify **roles="level-0 level-1 level-2"** to assign the level-0, level-1, and level-2 user roles to an HWTACACS user.

For information about configuring the device to cooperate with the remote AAA server for user role assignment, see AAA configuration in *Security Configuration Guide*.

If the AAA server assigns the security-audit user role and other user roles to the same user, only the security-audit user role takes effect.

If the AAA server assigns different admin roles (system-admin, security-admin, and audit-admin) to a user, only the first admin role takes effect.

Assigning user roles to local AAA authentication users

About this task

Configure user roles for local AAA authentication users in their local user accounts. For information about AAA and local user configuration, see AAA configuration in *Security Configuration Guide*.

Restrictions and guidelines

- Every local user has a default user role. If this default user role is not suitable, remove it.
- If a local user is the only user with the security-audit user role, the user cannot be deleted.
- The system-admin, security-admin, and audit-admin user roles are mutually exclusive with other user roles in a user account.
- The system-admin, security-admin, and audit-admin user roles are mutually exclusive in a user account.
- When you assign user roles to a user, the system prompts you to confirm the deletion of the user roles that are mutually exclusive with the new user roles.
- You can assign a maximum of 64 user roles to a local user.

Procedure

1. Enter system view.

`system-view`

2. Create a local user and enter its view.

`local-user` *`user-name`* **`class`** { **`manage`** | **`network`** }

3. Assign a user role to the local user.

`authorization-attribute user-role` *`role-name`*

The following default settings apply:

- The network-operator user role is assigned to local users created by a network-admin or level-15 user on the default context.
- The context-operator user role is assigned to local users created by a context-admin or level-15 user on a non-default context.

Assigning user roles to non-AAA authentication users on user lines

About this task

Specify user roles for the following two types of login users on the user lines:

- Non-SSH users that use password authentication or no authentication.
- SSH clients that use publickey or password-publickey authentication. User roles assigned to these SSH clients are specified in their respective device management user accounts.

For more information about user lines, see "Login overview" and "Configuring CLI login." For more information about SSH, see *Security Configuration Guide*.

Restrictions and guidelines

- You can assign a maximum of 64 user roles to a non-AAA authentication user on a user line.
- The device cannot assign the security-audit user role to non-AAA authentication users.
- The device cannot assign the following user roles to non-AAA authentication users:
    - system-admin.
    - security-admin.
    - audit-admin..

Procedure

1. Enter system view.

   `system-view`

2. Enter user line view or user line class view.
    - Enter user line view.

      line { *first-num1* [*last-num1*] | { **console** | **vty** } *first-num2* [*last-num2*] }
    - Enter user line class view.

      line class { **console** | **vty** }

   For information about the priority order and application scope of the settings in user line view and user line class view, see "Configuring CLI login."

3. Specify a user role on the user line.

   user-role *role-name*

   The following default settings apply:
    - The network-admin user role is specified on the console user line for default-context login users. The network-operator user role is specified on any other user line for default-context login users.
    - The network-admin user role of default-context login users changes to context-admin after the users use the **switchto context** command to log in to non-default contexts.
    - The context-operator user role is specified on user lines for other non-default context login users.

Configuring temporary user role authorization

About temporary user role authorization

Temporary user role authorization allows you to obtain another user role without reconnecting to the device. This feature is useful when you want to use a user role temporarily to configure a feature.

Temporary user role authorization is effective only on the current login. This feature does not change the user role settings in the user account that you have been logged in with. The next time you are logged in with the user account, the original user role settings take effect.

To enable a user to obtain another user role without reconnecting to the device, you must configure user role authentication. Table 2 describes the available authentication modes and configuration requirements.

Table 2 User role authentication modes

Keywords	Authentication mode	Description
local	Local password authentication only (local-only)	The device uses the locally configured password for authentication. If no local password is configured for a user role in this mode, a console user can obtain the user role by either entering a string or not entering anything.
scheme	Remote AAA authentication through HWTACACS or RADIUS (remote-only)	The device sends the username and password to the HWTACACS or RADIUS server for remote authentication. To use this mode, you must perform the following configuration tasks: • Configure the required HWTACACS or RADIUS scheme, and configure the ISP domain to use the scheme for the user. For more information, see AAA in *Security Configuration Guide*. • Add the user account and password on the HWTACACS or RADIUS server.
local scheme	Local password authentication first, and then remote AAA authentication (local-then-remote)	Local password authentication is performed first. If no local password is configured for the user role in this mode, the device performs remote AAA authentication for VTY users. A console user can obtain another user role by either entering a string or not entering anything.
scheme local	Remote AAA authentication first, and then local password authentication (remote-then-local)	Remote AAA authentication is performed first. Local password authentication is performed in either of the following situations: • The HWTACACS or RADIUS server does not respond. • The remote AAA configuration on the device is invalid.

Restrictions and guidelines for temporary user role authorization

If HWTACACS authentication is used, the following rules apply:

- If the device is not enabled to automatically obtain the login username as the authentication username, you must enter a username for role authentication.

- The device sends the username to the server in the *username or username@domain-name* format. Whether the domain name is included in the username depends on the **`user-name-format`** command in the HWTACACS scheme.
- To obtain a level-*n* user role, the user account on the server must have the target user role level or a level higher than the target user role. A user account that obtains the level-*n* user role can obtain any user role among level-0 through level-*n*.
- To obtain a non-level-*n* user role, make sure the user account on the server meets the following requirements:
  - The account has a user privilege level.
  - The HWTACACS custom attribute is configured for the account in the form of **allowed-roles="***role***"**. The variable *role* represents the target user role.

If RADIUS authentication is used, the following rules apply:

- The device does not use the username you enter or the automatically obtained login username to request user role authentication. It uses a username in the **$enab***n***$** format. The variable *n* represents a user role level, and a domain name is not included in the username. You can always pass user role authentication when the password is correct.
- To obtain a level-*n* user role, you must create a user account for the level-*n* user role in the **$enabn$** format on the RADIUS server. The variable *n* represents the target user role level. For example, to obtain the authorization of the level-3 user role, you can enter any username. The device uses the username **$enab3$** to request user role authentication from the server.
- To obtain a non-level-*n* user role, you must perform the following tasks:
  - Create a user account named **$enab0$** on the server.
  - Configure the cisco-av-pair attribute for the account in the form of **allowed-roles="***role***"**. The variable *role* represents the target user role.

The device selects an authentication domain for user role authentication in the following order:

1. The ISP domain included in the entered username.
2. The default ISP domain.

If you execute the **`quit`** command after obtaining user role authorization, you are logged out of the device.

Setting the authentication mode for temporary user role authorization

1. Enter system view.

   ```
   system-view
   ```

2. Set the authentication mode.

   ```
   super authentication-mode { local | scheme } *
   ```

   By default, local-only authentication applies.

Specifying the default target user role for temporary user role authorization

1. Enter system view.

   ```
   system-view
   ```

2. Specify the default target user role for temporary user role authorization.

   ```
   super default role role-name
   ```

   The following default settings apply:

- For default-context login users, the default target user role is network-admin.
- For non-default-context login users, the default target user role is context-admin.

Setting an authentication password for temporary user role authorization

About this task

Authentication passwords are required only for local password authentication.

Procedure

1. Enter system view.

   system-view

2. Set a local authentication password for a user role.

   super password [**role** *role-name*] [{ **hash** | **simple** } *string*]

   By default, no password is set.

   If you do not specify the **role** *role-name* option, the command sets a password for the default target user role.

Automatically obtaining the login username for temporary user role authorization

About this task

This feature is applicable only to the login from a user line that uses scheme authentication, which requires a username for login. This feature enables the device to automatically obtain the login username when the login user requests a temporary user role authorization from a remote authentication server.

Restrictions and guidelines

If the user was logged in from a user line that uses password authentication or no authentication, the device cannot obtain the login username. The request for temporary user role authorization from a remote authentication server will fail.

This feature does not take effect on local password authentication for temporary user role authorization.

Procedure

1. Enter system view.

   system-view

2. Enable the device to automatically obtain the login username when a login user requests temporary user role authorization from a remote authentication server.

   super use-login-username

   By default, the device prompts for a username when a login user requests temporary user role authorization from a remote authentication server.

Obtaining temporary user role authorization

Restrictions and guidelines

Enter the username (if any) and password within 60 seconds after you enter the **super** command. If you fail to do so, the command will time out. To obtain the role, you will need to re-execute the command.

The operation of obtaining temporary user role authorization fails after three consecutive unsuccessful authentication attempts.

You might fail to switch to a non-level-*n* user role if both of the following conditions exist:

- User role switching authentication is performed in the same ISP domain as the current login user.
- User role switching authentication uses a different AAA method than the login authorization method configured for the ISP domain.

To resolve this issue, make sure the AAA methods configured by using the **authentication super** command are consistent with those configured by using the **authorization login** command for the ISP domain.

For more information about AAA, see *Security Configuration Guide*.

Prerequisites

Before you obtain temporary user role authorization, make sure the current user account has the permission to execute the **super** command to obtain temporary user role authorization.

Procedure

To obtain the temporary authorization to use a user role, execute the following command in user view:

super [*role-name*]

If you do not specify the *role-name* argument, you obtain the default target user role for temporary user role authorization.

Display and maintenance commands for RBAC

Execute **display** commands in any view.

Task	Command
Display user role information.	**display role** [**name** *role-name*]
Display user role feature information.	**display role feature** [**name** *feature-name* | **verbose**]
Display user role feature group information.	**display role feature-group** [**name** *feature-group-name*] [**verbose**]

Troubleshooting RBAC

This section describes several typical RBAC issues and their solutions.

Local users have more access permissions than intended

Symptom

A local user can use more commands than should be permitted by the assigned user roles.

Analysis

The local user might have been assigned to user roles without your knowledge. For example, the local user is automatically assigned the default user role when you create the user.

Solution

To resolve the issue:

1. Use the **`display local-user`** command to examine the local user accounts for undesirable user roles, and remove them.
2. If the issue persists, contact NSFOCUS Support.

Login attempts by RADIUS users always fail

Symptom

Attempts by a RADIUS user to log in to the network access device always fail, even though the following conditions exist:

- The network access device and the RADIUS server can communicate with one another.
- All AAA settings are correct.

Analysis

RBAC requires that a login user have a minimum of one user role. If the RADIUS server does not authorize the login user to use any user role, the user cannot log in to the device.

Solution

To resolve the issue:

1. Use one of the following methods:
   - Configure the **`role default-role enable`** command. A RADIUS user can log in with the default user role when no user role is assigned by the RADIUS server.
   - Add the user role authorization attributes on the RADIUS server.
2. If the issue persists, contact NSFOCUS Support.

Contents

Login overview

The device supports the following types of login methods:

- **CLI login**—At the CLI, you can enter text commands to configure and manage the device.

  To log in to the CLI, you can use one of the following methods:

  - Connect to the console port.
  - Use Telnet.
  - Use SSH.
- **Web login**—Through the Web interface, you can configure and manage the device visually.
- **SNMP access**—You can run SNMP on an NMS to access the device MIB, and perform Get and Set operations to configure and manage the device.
- **RESTful access**—You can use RESTful API operations to configure and manage the device.

The first time you access the device, you can use one of the following methods:

- Connect to the console port and enter **admin** as both the username and password to log in to the CLI of the default context.
- Enter 192.168.0.1 in the address bar of the Web browser and enter **admin** as both the username and password to log in to the Web interface of the default context.

After login, you can change console or Web login parameters, configure other access methods, or create non-default contexts.

Non-default contexts do not have console ports. To log in to a non-default context for the first time, you must perform the following tasks:

1. Log in to the default context.
2. Switch to the non-default context by using the **`switchto context`** command.

After you log in to a non-default context, you can configure other access methods. Then, administrators of the default context and the non-default contexts can use the methods to access the non-default contexts. For more information about contexts, see *Virtual Technologies Configuration Guide*.

In login management related descriptions, it is assumed that the device does not enter the automatic configuration process at startup.

The first time you access the device, the default username and password are both **admin** for console login and Web login. The default IP address of the management interface on the device is 192.168.0.1/24.

Using the console port for the first device access

About this task

Console login is the fundamental login method.

Prerequisites

To log in through the console port, prepare a console terminal, for example, a PC. Make sure the console terminal has a terminal emulation program, such as HyperTerminal or PuTTY. For information about how to use terminal emulation programs, see the programs' user guides.

Procedure

1. Turn off the PC.

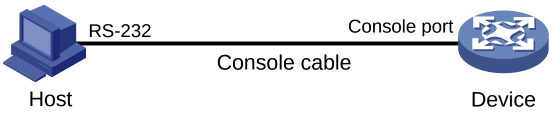
   The serial ports on PCs do not support hot swapping. Before connecting a cable to or disconnecting a cable from a serial port on a PC, you must turn off the PC.

2. Find the console cable shipped with the device and connect the DB-9 female connector of the console cable to the serial port of the PC.
3. Identify the console port of the device carefully and connect the RJ-45 connector of the console cable to the console port.

IMPORTANT:

To connect a PC to an operating device, first connect the PC end. To disconnect a PC from an operating device, first disconnect the device end.

Figure 1 Connecting a terminal to the console port



4. Turn on the PC.
5. On the PC, launch the terminal emulation program, and create a connection that uses the serial port connected to the device. Set the port properties so the port properties match the following console port default settings:
   - **Bits per second**—9600 bps.
   - **Flow control**—None.
   - **Parity**—None.
   - **Stop bits**—1.
   - **Data bits**—8.
6. Power on the device and press **Enter** as prompted.

   The user view prompt appears. You can enter commands to configure or manage the device. To get help, enter a question mark (?).

Configuring CLI login

About CLI login

The device uses user lines (also called user interfaces) to manage CLI sessions and monitor user behavior. For a user line, you can configure access control settings, including the login authentication method and user roles.

User lines

User line types

The device supports the following types of user lines, and different user lines require different login methods:

- **Console line**—Login through the console port.
- **Virtual type terminal (VTY) line**—Login through Telnet or SSH.

User line numbering

A user line has an absolute number and a relative number.

An absolute number uniquely identifies a user line among all user lines. The user lines are numbered starting from 0 and incrementing by 1, in the sequence of console and VTY lines. You can use the **`display line`** command without any parameters to view supported user lines and their absolute numbers.

A relative number uniquely identifies a user line among all user lines of the same type. The number format is *user line type* + *number*. All types of user lines are numbered starting from 0 and incrementing by 1. For example, the first VTY line is VTY 0.

User line assignment

The device assigns user lines to CLI login users depending on their login methods. When a user logs in, the device checks the idle user lines for the login method, and assigns the lowest numbered user line to the user. For example, if VTY 0 and VTY 3 are idle when a user Telnets to the device, the device assigns VTY 0 to the user.

Each user line can be assigned only to one user at a time. If no user line is available, a CLI login attempt will be rejected.

Login authentication modes

You can configure login authentication to prevent illegal access to the device CLI.

The device supports the following login authentication modes:

- **None**—Disables authentication. This mode allows access without authentication and is insecure.
- **Password**—Requires password authentication. A user must provide the correct password at login.
- **Scheme**—Uses the AAA module to provide local or remote login authentication. A user must provide the correct username and password at login.

Different login authentication modes require different user line configurations, as shown in Table 1.

Table 1 Configuration required for different login authentication modes

Authentication mode	Configuration tasks
None	Set the authentication mode to none.
Password	**1.** Set the authentication mode to password. **2.** Set a password.
Scheme	**1.** Set the authentication mode to scheme. **2.** Configure login authentication methods in ISP domain view. For more information, see *Security Configuration Guide*.

User roles

A user is assigned user roles at login. The user roles control the commands available for the user. For more information about user roles, see "Configuring RBAC."

The device assigns user roles based on the login authentication mode and user type.

- In none or password authentication mode, the device assigns the user roles specified for the user line.
- In scheme authentication mode, the device uses the following rules to assign user roles:
    - For an SSH login user who uses publickey or password-publickey authentication, the device assigns the user roles specified for the local device management user with the same name.
    - For other users, the device assigns user roles according to the user role configuration of the AAA module. If the AAA server does not assign any user roles and the default user role feature is disabled, a remote AAA authentication user cannot log in.

Restrictions and guidelines: CLI login configuration

For commands that are available in both user line view and user line class view, the following rules apply:

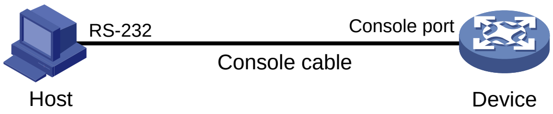
- A setting in user line view applies only to the user line. A setting in user line class view applies to all user lines of the class.
- A non-default setting in either view takes precedence over the default setting in the other view. A non-default setting in user line view takes precedence over the non-default setting in user line class view.
- A setting in user line class view takes effect only on users who log in after the setting is made. It does not affect users who are already online when the setting is made.

Configuring console login

About console login

You can connect a terminal to the console port of the device to log in and manage the device, as shown in Figure 2. For information about the login procedure, see "Using the console port for the first device access."

Figure 2 Logging in through the console port



By default, console login is enabled and the authentication mode is **`scheme`**. The username and password are both **admin**. The default user role is **network-admin** for a console user.

Restrictions and guidelines

A console login configuration change takes effect only on users who log in after the change is made. It does not affect users who are already online when the change is made.

Console login configuration tasks at a glance

To configure console login, perform the following tasks:

1. Configuring console login authentication
   - Disabling authentication for console login
   - Configuring password authentication for console login
   - Configuring scheme authentication for console login
2. (Optional.) Configuring common console login settings

Configuring console login authentication

Disabling authentication for console login

1. Enter system view.

   `system-view`

2. Enter console line view or class view.
   - Enter console line view.

     `line console` *`first-number`* `[` *`last-number`* `]`

   - Enter console line class view.

     `line class console`

3. Disable authentication.

   `authentication-mode none`

   By default, scheme authentication is enabled for console login.

CAUTION:

When authentication is disabled, users can log in to the device through the line or line class without authentication. For security purpose, disable authentication with caution.

4. Assign a user role.

   `user-role` *`role-name`*

   By default, a console user of the default context is assigned the **network-admin** user role. Non-default contexts do not support console login.

Configuring password authentication for console login

1. Enter system view.

system-view

2. Enter console line view or class view.
   - Enter Console line view.
     line console *first-number* [*last-number*]
   - Enter Console class view.
     line class console
3. Enable password authentication.
   authentication-mode password
   By default, scheme authentication is enabled for console login.
4. Set a password.
   set authentication password { **hash** | **simple** } *string*
   By default, no password is set.
5. Assign a user role.
   user-role *role-name*
   By default, a console user of the default context is assigned the **network-admin** user role. Non-default contexts do not support console login.

Configuring scheme authentication for console login

1. Enter system view.
   system-view
2. Enter console line view or class view.
   - Enter console line view.
     line console *first-number* [*last-number*]
   - Enter console line class view.
     line class console
3. Enable scheme authentication.
   authentication-mode scheme
   By default, scheme authentication is enabled for console login.

△ **CAUTION:**

When you enable scheme authentication, make sure an authentication user account is available. If no authentication user account is available, you cannot log in to the device through the line or line class at the next time.

4. Configure user authentication parameters in ISP domain view.
   To use local authentication, configure a local user and set the relevant attributes. To use remote authentication, configure a RADIUS, LDAP, or HWTACACS scheme. For more information, see AAA in *Security Configuration Guide*.

Configuring common console login settings

Restrictions and guidelines

Some common console login settings take effect immediately and can interrupt the current session. Use a login method different from console login to log in to the device before you change console login settings.

After you change console login settings, adjust the settings on the configuration terminal accordingly for a successful login.

Procedure

1. Enter system view.

   `system-view`

2. Enter console line view or class view.
   - Enter console line view.

     `line console` *`first-number`* `[` *`last-number`* `]`

   - Enter console line class view.

     `line class console`

3. Configure transmission parameters.
   - Set the transmission rate.

     `speed` *`speed-value`*

     By default, the transmission rate is 9600 bps.

     This command is not available in user line class view.

   - Specify the parity mode.

     `parity` `{` **`even`** `|` **`mark`** `|` **`none`** `|` **`odd`** `|` **`space`** `}`

     By default, a user line does not use parity.

     This command is not available in user line class view.

   - Configure flow control.

     `flow-control` `{` **`hardware`** `|` **`none`** `|` **`software`** `}`

     By default, the device does not perform flow control.

     This command is not available in user line class view.

   - Specify the number of data bits for a character.

     `databits` `{` **`7`** `|` **`8`** `}`

     The default is 8.

     This command is not available in user line class view.

Parameter	Description
7	Uses standard ASCII characters.
8	Uses extended ASCII characters.

   - Specify the number of stop bits for a character.

     `stopbits` `{` **`1`** `|` **`1.5`** `|` **`2`** `}`

     The default is 1.

     Stop bits indicate the end of a character. The more the stop bits, the slower the transmission.

     This command is not available in user line class view.

4. Configure terminal attributes.
   - Enable the terminal service.

     `shell`

     Be default, the terminal service is enabled on all user lines.

     The **`undo shell`** command is not available in console line view.

   - Specify the terminal display type.

     `terminal type` `{` **`ansi`** `|` **`vt100`** `}`

     By default, the terminal display type is ANSI.

The device supports ANSI and VT100 terminal display types. As a best practice, specify VT100 type on both the device and the configuration terminal. You can also specify the ANSI type for both sides, but a display problem might occur if a command line has more than 80 characters.

- Set the maximum number of lines of command output to send to the terminal at a time.

  `screen-length` *`screen-length`*

  By default, the device sends a maximum of 24 lines to the terminal at a time.

  To disable pausing between screens of output, set the value to 0.

- Set the size for the command history buffer.

  `history-command max-size` *`value`*

  By default, the buffer size is 10. The buffer for a user line can save a maximum of 10 history commands.

- Set the CLI connection idle-timeout timer.

  `idle-timeout` *`minutes`* `[` *`seconds`* `]`

  By default, the CLI connection idle-timeout timer is 10 minutes.

  If no interaction occurs between the device and the user within the idle-timeout interval, the system automatically terminates the user connection on the user line.

  If you set the timeout timer to 0, the connection will not be aged out.

5. Specify the command to be automatically executed for login users on the lines.

`auto-execute command` *`command`*

By default, no command is specified for auto execution.

△ **CAUTION:**

Use this command with caution. If this command is used on a user line, users that log in to the device through this user line might fail to configure the system.

The device will automatically execute the specified command when a user logs in through the user line, and close the user connection after the command is executed.

This command is not available in console line view or console line class view.

6. Configure shortcut keys.

- Specify the terminal session activation key.

  `activation-key` *`character`*

  By default, pressing **Enter** starts the terminal session.

- Specify the escape key.

  `escape-key` { *`key-string`* | **`default`** }

  By default, pressing **Ctrl+C** terminates a command.

- Set the user line locking key.

  `lock-key` *`key-string`*

  By default, no user line locking key is set.

Configuring Telnet login

About Telnet login

The device can act as a Telnet server to allow Telnet login, or as a Telnet client to Telnet to other devices.

Restrictions and guidelines

A Telnet login configuration change takes effect only on users who log in after the change is made. It does not affect users who are already online when the change is made.

Configuring the device as a Telnet server

Telnet server configuration tasks at a glance

To configure the device as a Telnet server, perform the following tasks:

1. Enabling the Telnet server
2. Configuring Telnet login authentication
   - Disabling authentication for Telnet login
   - Configuring password authentication for Telnet login
   - Configuring scheme authentication for Telnet login
3. (Optional.) Configuring common Telnet server settings
4. (Optional.) Configuring common VTY line settings

Enabling the Telnet server

1. Enter system view.

   `system-view`

2. Enable the Telnet server.

   `telnet server enable`

   By default, the Telnet server is disabled.

Disabling authentication for Telnet login

1. Enter system view.

   `system-view`

2. Enter VTY line view or class view.
   - Enter VTY line view.

     `line vty` *`first-number`* `[` *`last-number`* `]`

   - Enter VTY line class view.

     `line class vty`

3. Disable authentication.

   `authentication-mode none`

   By default, scheme authentication is enabled for Telnet login.

   > △ **CAUTION:**
   >
   > When authentication is disabled, users can log in to the device through the line or line class without authentication. For security purpose, disable authentication with caution.

   In VTY line view, this command is associated with the **`protocol inbound`** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

4. (Optional.) Assign a user role.

   `user-role` *`role-name`*

   By default, a Telnet user of the default context is assigned the **network-operator** user role, and a Telnet user of a non-default context is assigned the **context-operator** user role.

Configuring password authentication for Telnet login

1. Enter system view.

   `system-view`

2. Enter VTY line view or class view.
   - Enter VTY line view.

     `line vty` *`first-number`* `[` *`last-number`* `]`
   - Enter VTY line class view.

     `line class vty`

3. Enable password authentication.

   `authentication-mode password`

   By default, scheme authentication is enabled for Telnet login.

   > △ **CAUTION:**
   >
   > When you enable password authentication, you must also configure an authentication password for the line or line class. If no authentication password is configured, you cannot log in to the device through the line or line class at the next time.

   In VTY line view, this command is associated with the **`protocol inbound`** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

4. Set a password.

   `set authentication password` `{` **`hash`** `|` **`simple`** `}` *`password`*

   By default, no password is set.

5. (Optional.) Assign a user role.

   `user-role` *`role-name`*

   By default, a Telnet user of the default context is assigned the **network-operator** user role, and a Telnet user of a non-default context is assigned the **context-operator** user role.

Configuring scheme authentication for Telnet login

1. Enter system view.

   `system-view`

2. Enter VTY line view or class view.
   - Enter VTY line view.

     `line vty` *`first-number`* `[` *`last-number`* `]`
   - Enter VTY line class view.

     `line class vty`

3. Enable scheme authentication.

   `authentication-mode scheme`

   By default, scheme authentication is enabled for Telnet login.

   > △ **CAUTION:**
   >
   > When you enable scheme authentication, make sure an authentication user account is available. If no authentication user account is available, you cannot log in to the device through the line or line class at the next time.

   In VTY line view, this command is associated with the **`protocol inbound`** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

4. Configure user authentication parameters in ISP domain view.

   To use local authentication, configure a local user and set the relevant attributes.

   To use remote authentication, configure a RADIUS, LDAP, or HWTACACS scheme. For more information, see AAA in *Security Configuration Guide*.

Configuring common Telnet server settings

1. Enter system view.

   `system-view`

2. Set the DSCP value for outgoing Telnet packets.

   IPv4:

   `telnet server dscp` *`dscp-value`*

   IPv6:

   `telnet server ipv6 dscp` *`dscp-value`*

   By default, the DSCP value is 48.

3. Specify the Telnet service port number.

   IPv4:

   `telnet server port` *`port-number`*

   IPv6:

   `telnet server ipv6 port` *`port-number`*

   By default, the Telnet service port number is 23.

4. Set the maximum number of concurrent Telnet users.

   `aaa session-limit telnet` *`max-sessions`*

   By default, the maximum number of concurrent Telnet users is 32.

   Changing this setting does not affect users who are currently online. If the new limit is less than the number of online Telnet users, no additional users can Telnet in until the number drops below the new limit.

   For more information about this command, see *Security Command Reference*.

Configuring common VTY line settings

1. Enter system view.

   `system-view`

2. Enter VTY line view or class view.
   - Enter VTY line view.

     `line vty` *`first-number`* `[` *`last-number`* `]`
   - Enter VTY line class view.

     `line class vty`

3. Configure VTY terminal attributes.
   - Enable the terminal service.

     `shell`

     By default, the terminal service is enabled on all user lines.
   - Specify the terminal display type.

     `terminal type` `{` **`ansi`** `|` **`vt100`** `}`

     By default, the terminal display type is ANSI.
   - Set the maximum number of lines of command output to send to the terminal at a time.

     `screen-length` *`screen-length`*

     By default, the device sends a maximum of 24 lines to the terminal at a time.

To disable pausing between screens of output, set the value to 0.

- Set the size for the command history buffer.

  history-command max-size *value*

  By default, the buffer size is 10. The buffer for a user line can save a maximum of 10 history commands.

- Set the CLI connection idle-timeout timer.

  idle-timeout *minutes* [*seconds*]

  By default, the CLI connection idle-timeout timer is 10 minutes.

  If no interaction occurs between the device and the user within the idle-timeout interval, the system automatically terminates the user connection on the user line.

  If you set the timeout timer to 0, the connection will not be aged out.

4. Specify the supported protocols.

   protocol inbound { **all** | **ssh** | **telnet** }

   By default, Telnet and SSH are supported.

   A protocol change takes effect only on users who log in after the setting is made. It does not affect users who are already online when the setting is made.

   In VTY line view, this command is associated with the **authentication-mode** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

5. Specify the command to be automatically executed for login users on the user lines.

   auto-execute command *command*

   By default, no command is specified for auto execution.

   IMPORTANT:

   Before you use this command and save the configuration, make sure you can access the CLI to modify the configuration through other VTY lines.

   For a VTY line, you can specify a command that is to be automatically executed when a user logs in. After executing the specified command, the system automatically disconnects the Telnet session.

6. Configure shortcut keys.

   - Specify the shortcut key for terminating a task.

     escape-key { *character* | **default** }

     The default setting is **Ctrl+C**.

   - Set the user line locking key.

     lock-key *key-string*

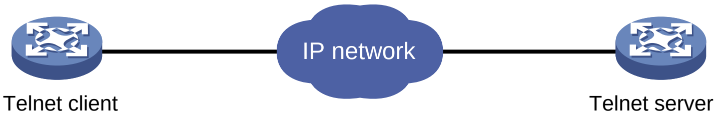
     By default, no user line locking key is set.

Using the device to log in to a Telnet server

About this task

You can use the device as a Telnet client to log in to a Telnet server.

Figure 3 Telnetting from the device to a Telnet server



Prerequisites

Assign an IP address to the device and obtain the IP address of the Telnet server. If the device resides on a different subnet than the Telnet server, make sure the device and the Telnet server can reach each other.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Specify the source IPv4 address or source interface for outgoing Telnet packets.

   telnet client source { **interface** *interface-type interface-number* | **ip** *ip-address* }

   By default, no source IPv4 address or source interface is specified. The device uses the primary IPv4 address of the output interface as the source address for outgoing Telnet packets.

3. Return to user view.

   quit

4. Use the device to log in to a Telnet server.

   IPv4:

   telnet *remote-host* [*service-port*] [**vpn-instance** *vpn-instance-name*] [**source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] [**dscp** *dscp-value*] [**escape** *character*]

   IPv6:

   telnet ipv6 *remote-host* [**-i** *interface-type interface-number*] [*port-number*] [**vpn-instance** *vpn-instance-name*] [**source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] [**dscp** *dscp-value*] [**escape** *character*]

Configuring SSH login

About SSH login

SSH offers a secure remote login method. By providing encryption and strong authentication, it protects devices against attacks such as IP spoofing and plaintext password interception. For more information, see *Security Configuration Guide*.

The device can act as an SSH server to allow Telnet login, or as an SSH client to log in to an SSH server.

Configuring the device as an SSH server

About this task

This section provides the SSH server configuration procedure used when the SSH client authentication method is password. For more information about SSH and publickey authentication configuration, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create local key pairs.

   public-key local create { **dsa** | **ecdsa secp256r1** | **rsa** }

3. Enable the SSH server.

   ssh server enable

   By default, the SSH server is enabled.

4. (Optional.) Create an SSH user and specify the authentication mode.

   ssh user *username* **service-type stelnet authentication-type password**

5. Enter VTY line view or class view.
   - Enter VTY line view.

     line vty *first-number* [*last-number*]
   - Enter VTY line class view.

     line class vty

6. Enable scheme authentication.

   authentication-mode scheme

   By default, scheme authentication is enabled for VTY lines.

   In VTY line view, this command is associated with the **protocol inbound** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

△ **CAUTION:**

When you enable scheme authentication, make sure an authentication user account is available. If no authentication user account is available, you cannot log in to the device through the line or line class at the next time.

7. (Optional.) Specify the protocols for the user lines to support.

   protocol inbound { **all** | **ssh** | **telnet** }

   By default, Telnet and SSH are supported.

   A protocol change takes effect only on users who log in after the setting is made. It does not affect users who are already online when the setting is made.

   In VTY line view, this command is associated with the **authentication-mode** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

8. (Optional.) Set the maximum number of concurrent SSH users.

   aaa session-limit ssh *max-sessions*

   By default, the maximum number of concurrent SSH users is 32.

   Changing this setting does not affect users who are currently online. If the new limit is less than the number of online SSH users, no additional SSH users can log in until the number drops below the new limit.

   For more information about this command, see *Security Command Reference*.

9. (Optional.) Configure common settings for VTY lines:

   a. Return to system view.

      quit

   b. Configure common settings for VTY lines.

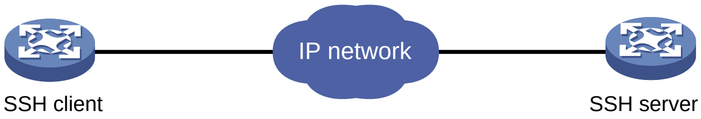
See "Configuring common VTY line settings."

Using the device to log in to an SSH server

About this task

You can use the device as an SSH client to log in to an SSH server.

Figure 4 Logging in to an SSH server from the device



Prerequisites

Assign an IP address to the device and obtain the IP address of the SSH server. If the device resides on a different subnet than the SSH server, make sure the device and the SSH server can reach each other.

Procedure

To use the device to log in to an SSH server, execute one of the following commands in user view:

IPv4:

ssh2 *server*

IPv6:

ssh2 ipv6 *server*

To work with the SSH server, you might need to specify a set of parameters. For more information, see *Security Configuration Guide*.

Display and maintenance commands for CLI login

Execute **display** commands in any view.

Task	Command	Remarks
Display user line information.	**display line** [*num1* | { **console** | **vty** } *num2*] [**summary**]	N/A
Display the packet source setting for the Telnet client.	**display telnet client**	N/A
Display online CLI users.	**display users** [**all**]	N/A
Release a user line.	**free line** { *num1* | { **console** | **vty** } *num2* }	Multiple users can log in to the device to simultaneously configure the device. When necessary, you can execute this command to release some connections. You cannot use this command to release the connection you are using. This command is available in user view.
Lock the current user line and set the password for unlocking the line.	**lock**	By default, the system does not lock any user lines.

Task	Command	Remarks
		This command is available in user view.
Lock the current user line and enable unlocking authentication.	**lock reauthentication**	By default, the system does not lock any user lines or initiate reauthentication. To unlock the locked user line, you must press **Enter** and provide the login password to pass reauthentication. This command is available in any view.
Send messages to user lines.	**send** { **all** | *num1* | { **console** | **vty** } *num2* }	This command is available in user view.

Configuring Web login

About Web login

The device provides a built-in Web server that supports HTTP 1.0, HTTP 1.1, and HTTPS. You can use a Web browser to log in to and configure the device.

HTTPS uses SSL to ensure the integrity and security of data exchanged between the client and the server, and is more secure than HTTP. You can define a certificate-based access control policy to allow only legal clients to access the Web interface.

IMPORTANT:

On a PC, a Web browser allows only one user to log in to the device. If multiple users use the same Web browser to log in to the device, the most recent login takes effect.

Restrictions and guidelines: Web login configuration

To improve device security, the system automatically enables the HTTPS service when you enable the HTTP service. When the HTTP service is enabled, you cannot disable the HTTPS service.

Web login configuration tasks at a glance

To configure Web login, perform the following tasks:

1. Configuring Web login
   - Configuring HTTP login
   - Configuring HTTPS login
2. Configuring a Web login local user
3. Managing Web connections

Prerequisites for Web login

Before logging in to the Web interface of the device, log in to the device by using any other method and assign an IP address to the device. Make sure the configuration terminal and the device can communicate over the IP network.

Configuring HTTP login

1. (Optional.) Specify a fixed verification code for Web login.

   ```
   web captcha verification-code
   ```

   By default, no fixed verification code is specified. A Web user must enter the verification code displayed on the login page at login.

   Execute this command in user view.

2. Enter system view.

   ```
   system-view
   ```

3. Enable the HTTP service.

   ```
   ip http enable
   ```

   By default, the HTTP service is disabled.

4. (Optional.) Specify the HTTP service port number.

   ```
   ip http port port-number
   ```

   The default HTTP service port number is 80.

5. (Optional.) Specify the HTTP methods to be added to the reply to an OPTIONS request.

   ```
   http method { delete | get | head | options | post | put } *
   ```

   By default, no HTTP methods are specified.

Configuring HTTPS login

About this task

The device supports the following HTTPS login modes:

- **Simplified mode**—Using this mode is easy in configuration but has low security. The device uses a self-signed certificate (a certificate that is generated and signed by the device itself) and the default SSL settings. The device operates in simplified mode after you enable HTTPS service on the device. You do not need to configure an SSL server policy with the HTTPS service. However, a self-signed certificate is not signed by a CA, so it is not trusted by a browser. The browser will prompt a security risk message when you access the device through HTTPS. If you do not have high security requirements and can accept the security risk with a self-signed certificate, you can choose to ignore this message and continue browsing the webpage.
- **Secure mode**—Using this mode is complicated in configuration but has high security. The device uses a certificate signed by a CA and a set of user-defined security protection settings to ensure security. For the device to operate in secure mode, you must perform the following tasks:
  - Obtain a CA certificate and apply a local certificate from the CA.
  - Enable HTTPS service on the device.
  - Specify an SSL server policy for the service.
  - Configure PKI domain-related parameters.

IMPORTANT:

To use secure mode for HTTPS login, first purchase a local certificate for SSL from an official third-party CA organization. NSFOCUS does not provide the device with a CA certificate issued by an authoritative organization.

For more information about SSL, self-signed certificates, CA-signed certificates, and PKI, see *Security Configuration Guide*.

Restrictions and guidelines

- If the HTTPS service and the SSL VPN service use the same port number, they must use the same SSL server policy. If they use different SSL server policies, only one of them can be enabled.

  To modify the SSL server policy used by both the HTTPS service and the SSL VPN service, you must perform the following tasks:

  a. Disable the two services before you modify the SSL server policy.

  b. Enable the two services again after the modification.

  If you fail to complete the required tasks, the new settings do not take effect.

- To associate a different SSL server policy with the HTTPS service, you must perform the following tasks:
  a. Disable the HTTP service and HTTPS service before you associate the new SSL server policy.
  b. Enable the HTTP service and HTTPS service again after the association.

  If you fail to complete the required tasks, the new SSL server policy does not take effect.
- For the HTTP service to use its self-signed certificate after you associate an SSL server policy with the HTTPS service, you must follow these steps:
  a. Disable the HTTP service and HTTPS service.
  b. Execute the **`undo ip https ssl-server-policy`** command to remove the existing SSL server policy association.
  c. Enable the HTTP service and HTTPS service again.
- Enabling the HTTPS service triggers the SSL handshake negotiation process.
  - If the device has a local certificate, the SSL handshake negotiation succeeds and the HTTPS service starts up.
  - If the device does not have a local certificate, the certificate application process starts. Because the certificate application process takes a long time, the SSL handshake negotiation might fail and the HTTPS service might not be started. To solve the problem, execute the **`ip https enable`** command again until the HTTPS service is enabled.
- To use a certificate-based access control policy to control HTTPS access, you must perform the following tasks:
  - Use the **`client-verify enable`** command in the SSL server policy that is associated with the HTTPS service.
  - Configure a minimum of one **permit** rule in the certificate-based access control policy.

  If you fail to complete the required tasks, HTTPS clients cannot log in.

Procedure

1. (Optional.) Specify a fixed verification code for Web login.

`web captcha` *`verification-code`*

By default, no fixed verification code is configured. A Web user must enter the verification code displayed on the login page at login.

2. Enter system view.

`system-view`

3. (Optional.) Apply policies to the HTTPS service.

- Apply an SSL server policy.

  `ip https ssl-server-policy` *`policy-name`*

  By default, no SSL server policy is associated. The HTTP service uses a self-signed certificate.
- Apply a certificate-based access control policy to control HTTPS access.

  `ip https certificate access-control-policy` *`policy-name`*

  By default, no certificate-based access control policy is applied.

  For more information about certificate-based access control policies, see PKI in *Security Configuration Guide*.

4. Enable the HTTPS service.

`ip https enable`

By default, the HTTPS service is enabled.

5. (Optional.) Specify the HTTPS service port number.

ip https port *port-number*

The default HTTPS service port number is 443.

6. (Optional.) Set the authentication and authorization mode for HTTPS login.

   web https-authorization mode { **auto** | **certificate** | **certificate-manual** | **manual** }

   By default, manual mode is used for HTTPS login authentication and authorization.

7. (Optional.) Specify the certificate field to be used as the username for certificate-based authentication.

   web https-authorization username { **cn** | **email-prefix** | **oid** *oid-value* }

   By default, the CN field in the certificate is used as the username for certificate-based authentication.

Configuring a Web login local user

1. Enter system view.

   system-view

2. Create a local user and enter local user view.

   local-user *user-name* [**class manage**]

3. (Optional.) Configure a password for the local user.

   password [{ **hash** | **simple** } *password*]

   By default, no password is configured for a local user. The local user can pass authentication after entering the correct username and passing attribute checks.

4. Configure user attributes.

   - Assign a user role to the local user.

     authorization-attribute user-role *user-role*

     The default user role is network-operator for a Web user.

   - Specify the service type for the local user.

     service-type { **http** | **https** }

     By default, no service type is specified for a local user.

Managing Web connections

Setting the Web connection idle-timeout timer

1. Enter system view.

   system-view

2. Set the Web connection idle-timeout timer.

   web idle-timeout *minutes*

   By default, the Web connection idle-timeout timer is 10 minutes.

Specifying the maximum number of online HTTP or HTTPS users

1. Enter system view.

   system-view

2. Specify the maximum number of online HTTP or HTTPS users.

   aaa session-limit { **http** | **https** } *max-sessions*

By default, the device supports a maximum number of 32 online HTTP users and 32 online HTTPS users.

Changing this setting does not affect users who are currently online. If the new setting is less than the number of online HTTP or HTTPS users, no additional HTTP or HTTPS users can log in until the number drops below the new limit. For more information about this command, see *Security Command Reference*.

Logging off Web users

To log off Web users, execute the following command in user view:

free web users { **all** | **user-id** *user-id* | **user-name** *user-name* }

Display and maintenance commands for Web login

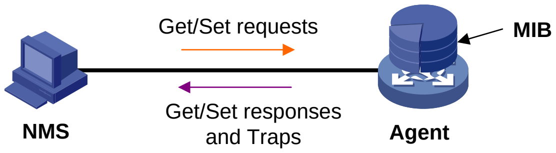
Execute **display** commands in any view and the **free web users** command in user view.

Task	Command
Display HTTP service configuration and status information.	**display ip http**
Display HTTPS service configuration and status information.	**display ip https**
Display Web interface navigation tree information.	**display web menu** [**chinese**]
Display online Web users.	**display web users**
Log off online Web users.	**free web users** { **all** | **user-id** *user-id* | **user-name** *user-name* }

Accessing the device through SNMP

You can run SNMP on an NMS to access the device MIB and perform Get and Set operations to configure and manage the device.

Figure 5 SNMP access diagram



For more information about SNMP, see *Network Management and Monitoring Configuration Guide*.

Configuring RESTful access

About RESTful access

The device provides the Representational State Transfer application programming interface (RESTful API). Based on this API, you can use programming languages such as Python, Ruby, or Java to write programs to perform the following tasks:

- Send RESTful requests to the device to pass authentication.
- Use RESTful API operations to configure and manage the device. RESTful API operations include Get, Put, Post, and Delete.

The device supports using HTTP or HTTPS to transfer RESTful packets.

Configuring RESTful access over HTTP

1. Enter system view.

   system-view

2. (Optional.) Specify the service port number for RESTful access over HTTP.

   restful http port *port-number*

   By default, the service port number for RESTful access over HTTP is 80.

3. Enable RESTful access over HTTP.

   restful http enable

   By default, RESTful access over HTTP is disabled.

4. Create a local user and enter local user view.

   local-user *user-name* [**class manage**]

5. Configure a password for the local user.

   password [{ **hash** | **simple** } *password*]

6. (Optional.) Assign a user role to the local user.

   authorization-attribute user-role *user-role*

   The default user role is network-operator for a RESTful access user.

7. Specify the HTTP service for the local user.

   service-type http

   By default, no service type is specified for a local user.

Configuring RESTful access over HTTPS

1. Enter system view.

   system-view

2. (Optional.) Apply an SSL server policy to the RESTful access over HTTPS service.

   restful https ssl-server-policy *policy-name*

   By default, no SSL server policy is applied to the RESTful access over HTTPS service.

   The RESTful access over HTTPS service will use the SSL server policy to enhance service security. For more information about SSL server policies, see SSL configuration in *Security Configuration Guide*.

3. (Optional.) Specify the service port number for RESTful access over HTTPS.

   restful https port *port-number*

   By default, the service port number for RESTful access over HTTPS is 443.

4. Enable RESTful access over HTTPS.

   restful https enable

   By default, RESTful access over HTTPS is disabled.

5. Create a local user and enter local user view.

   local-user *user-name* [**class manage**]

6. Configure a password for the local user.

   password [{ **hash** | **simple** } *password*]

7. (Optional.) Assign a user role to the local user.

   authorization-attribute user-role *user-role*

   The default user role is network-operator for a RESTful access user.

8. Specify the HTTPS service for the local user.

   service-type https

   By default, no service type is specified for a local user.

Controlling user access to the device

About login user access control

Use ACLs to prevent unauthorized access, and configure command authorization and accounting to monitor and control user behavior.

To control user access, specify an ACL that has rules so only users permitted by the ACL can access the device.

- If no ACL is applied, all users can access the device.
- If the ACL for Web user access control does not exist or does not have rules, all Web users can access the device.
- If the ACL for Telnet, SSH, or SNMP access control does not exist or does not have rules, no Telnet, SSH, or SNMP users can access the device.
- If a VPN instance is specified in an ACL rule, the rule applies only to the packets of the VPN instance. If no VPN instance is specified in an ACL rule, the rule applies only to the packets on the public network.

For more information about ACLs, see *ACL and QoS Configuration Guide.*

Controlling Telnet and SSH logins

Controlling Telnet logins

1. Enter system view.

   `system-view`

2. Apply an ACL to control Telnet logins.

   IPv4:

   `telnet server acl` `[` **`mac`** `]` *`acl-number`*

   IPv6:

   `telnet server ipv6 acl` `{` **`ipv6`** `|` **`mac`** `}` *`acl-number`*

   By default, no ACL is used to control Telnet logins.

3. (Optional.) Enable logging for Telnet login attempts that are denied by the Telnet login control ACL.

   `telnet server acl-deny-log enable`

   By default, logging is disabled for Telnet login attempts that are denied by the Telnet login control ACL.

Controlling SSH logins

1. Enter system view.

   `system-view`

2. Apply an ACL to control SSH logins.

   IPv4:

   `ssh server acl` `{` *`advanced-acl-number`* `|` *`basic-acl-number`* `|` **`mac`** *`mac-acl-number`* `}`

IPv6:

```
ssh server ipv6 acl { ipv6 { advanced-acl-number | basic-acl-number }
| mac mac-acl-number }
```

By default, no ACL is used to control SSH logins.

3. (Optional.) Enable logging for SSH login attempts that are denied by the SSH login control ACL.

```
ssh server acl-deny-log enable
```

By default, logging is disabled for SSH login attempts that are denied by the SSH login control ACL.

For more information about **ssh** commands, see SSH in *Security Command Reference*.

Controlling Web logins

Configuring source IP-based Web login control

1. Enter system view.

```
system-view
```

2. Apply a basic ACL to control Web logins.
   - Control HTTP logins.

```
ip http acl [ ipv6 ] [ advanced ] { acl-number | name acl-name }
ip http acl mac { acl-number | name acl-name }
```

   - Control HTTPS logins.

```
ip https acl [ ipv6 ] [ advanced ] { acl-number | name acl-name }
ip https acl mac { acl-number | name acl-name }
```

By default, no ACL is applied to control Web logins.

Configuring command authorization

About command authorization

By default, commands available for a user depend only on the user's user roles. When the authentication mode is scheme, you can configure the command authorization feature to further control access to commands.

After you enable command authorization, a user can use only commands that are permitted by both the AAA scheme and user roles.

Restrictions and guidelines

When command authorization is enabled, commands available for a user vary by the user's login authentication mode.

- If authentication is disabled or password authentication is enabled, command authorization does not take effect, and the user cannot use any commands.
- If scheme authentication is enabled, commands available for a user vary by the user's access authentication method.
  - If local authentication is used, the device uses the user roles assigned to the user to perform command authorization.

- If remote authentication is used, the remote authorization server performs command authorization to determine whether a command entered by a login user is permitted. If remote authorization fails, the device uses the user roles of a local user with the same name as the login user to determine whether the command can be used. If the authorization also fails, the login user cannot use the command.

Command authorization configuration changes in user line class view do not take effect on the current session. The changes take effect only on subsequent login sessions. Command authorization configuration changes in user line view take effect immediately on all users that access the user line.

If the remote server performs command authorization, you must configure a command authorization method in ISP domain view. The command authorization method can be different from the user login authorization method. For more information, see AAA in *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter user line view or user line class view.

   - Enter user line view.

     `line` { *`first-number1`* [*`last-number1`*] | { **`console`** | **`vty`** } *`first-number2`* [*`last-number2`*] }

   - Enter user line class view.

     `line class` { **`console`** | **`vty`** }

   A setting in user line view applies only to the user line. A setting in user line class view applies to all user lines of the class. A non-default setting in either view takes precedence over the default setting in the other view. A non-default setting in user line view takes precedence over the non-default setting in user line class view.

   A setting in user line class view takes effect only on users who log in after the setting is made. It does not affect users who are already online when the setting is made.

3. Enable scheme authentication.

   `authentication-mode scheme`

   By default, scheme authentication is enabled for console login and VTY login.

   In VTY line view, this command is associated with the **`protocol inbound`** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

   △ **CAUTION:**

   When you enable scheme authentication, make sure an authentication user account is available. If no authentication user account is available, you cannot log in to the device through the line or line class at the next time.

4. Enable command authorization.

   `command authorization`

   By default, command authorization is disabled, and the commands available for a user only depend on the user role.

   If the **`command authorization`** command is used in user line class view, command authorization is enabled on all user lines in the class. You cannot use the **`undo command authorization`** command in the view of a user line in the class.

Configuring command accounting

About command accounting

Command accounting uses the HWTACACS server to record all executed commands to monitor user behavior on the device.

If command accounting is enabled but command authorization is not, every executed command is recorded. If both command accounting and command authorization are enabled, only authorized commands that are executed are recorded.

Restrictions and guidelines

The command accounting method can be the same as or different from the command authorization method and user login authorization method.

For the command accounting feature to take effect, you must configure a command accounting method in ISP domain view. For more information, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter user line view or user line class view.
   - Enter user line view.

     ```
     line { first-number1 [ last-number1 ] | { console | vty }
     first-number2 [ last-number2 ] }
     ```

   - Enter user line class view.

     ```
     line class { console | vty }
     ```

   A setting in user line view applies only to the user line. A setting in user line class view applies to all user lines of the class. A non-default setting in either view takes precedence over the default setting in the other view. A non-default setting in user line view takes precedence over the non-default setting in user line class view.

   A setting in user line class view takes effect only on users who log in after the setting is made. It does not affect users who are already online when the setting is made.

3. Enable scheme authentication.

   ```
   authentication-mode scheme
   ```

   By default, scheme authentication is enabled for console login and VTY login.

   In VTY line view, this command is associated with the **protocol inbound** command. If one command has a non-default setting in VTY line view, the other command uses its setting in VTY line view, regardless of its setting in VTY line class view.

> △ **CAUTION:**
>
> When you enable scheme authentication, make sure an authentication user account is available. If no authentication user account is available, you cannot log in to the device through the line or line class at the next time.

4. Enable command accounting.

   ```
   command accounting
   ```

By default, command accounting is disabled. The accounting server does not record the commands executed by users.

If the **command accounting** command is used in user line class view, command accounting is enabled on all user lines in the class. You cannot use the **undo command accounting** command in the view of a user line in the class.

Contents

Managing licenses

About licenses

To obtain information about license-based features, their licensing status, and license availability, execute the **`display license feature`** command on the device. Then, you can purchase and install licenses as needed.

License types

NSFOCUS offers trial (or temporary), preinstalled, and formal licenses. For more information about the license options available for a product, see the license matrixes document for that product.

Preinstalled licenses

Preinstalled licenses are built into a system and are available for use on initial startup of the system. Users do not need to activate them.

A preinstalled license is typically time limited and cannot be uninstalled or transferred.

When a preinstalled license expires, the license-based feature becomes unavailable. To continue to use the feature, you must purchase and install a formal license for it.

Trial licenses

Trial licenses are provided for you to verify the functionality of premium features so you can make an educated purchase decision.

A trial license is time limited and cannot be uninstalled or transferred.

When a trial license expires, the license-based feature becomes unavailable. To continue to use the feature, you must purchase and install a formal license for it.

To obtain a trial license for a feature:

1. Contact your NSFOCUS sales representative or technical support.to obtain a trial license key and an activation file.
2. Install the activation file to activate the feature.

Formal licenses

To gain most from a license-based feature, purchase a formal license for it.

Formal licenses are available with different validity periods and their support for uninstallation and transfer of formal licenses depends on product model. For more information, see the license matrixes document for the product.

To install a formal license for a feature:

1. Purchase a software license certificate through an official channel.
2. Provide the license key in the software license certificate to NSFOCUS sales representative or technical support to obtain an activation file.
3. Install the activation file in the target system.

Basic concepts

The following information describes the basic concepts that you might encounter when you register, install, and manage licenses.

Software license certificate

A software license certificate allows users to use a license-based feature. It contains license key, license capacity, and other information.

License key

A license key uniquely identifies a license.

- To obtain a formal license key, purchase a software license certificate. The authorization serial number in the software license certificate is the license key.
- To obtain a trial license key, contact your NSFOCUS sales representative or NSFOCUS technical support. Support for trial licenses depends on the device model. For more information, contact NSFOCUS sales representative or NSFOCUS technical support.

Device serial number

A device serial number (SN or S/N) is a barcode that uniquely identifies a device. It comes with the device and must be provided when you contact NSFOCUS to request a license.

Device ID (DID) and DID file

A DID is a string of characters that uniquely identifies a hardware device. A DID file stores the DID and other information. The device comes with a DID or DID file. You must provide the DID or DID file when you contact NSFOCUS to request a license.

Activation file

An activation file binds a license to a system.

To use a license-based feature on a system, you must perform the following tasks:

1. Use the license key and the required device information to obtain an activation file from NSFOCUS.
2. Install the activation file on the system.

Uninstall key and Uninstall file

When you uninstall a license, an Uninstall file that contains an Uninstall key is created. The Uninstall key is required for transferring the license.

License storage

License storage is a persistent storage of fixed size for storing licensing information. This information includes the licensing state, validity period, Uninstall key or Uninstall file, and other related information.

Data in the license storage persists through reboot. This ensures licensing accuracy and continuity.

Restrictions and guidelines: License management

Management operation restrictions

- Purchase licenses from NSFOCUS official channels.
- For licenses that have been installed on the device, execute the **`display license`** command to view the license validity period. To use a license-based feature continuously, install a new license for the feature before the old license expires.
- Licenses are typically device locked. To ensure a successful licensing, use the following licensing guidelines:
  - **a.** When you purchase a license certificate, verify the following items:
    - Make sure the license is compatible with the target device.

    - Make sure its licensed functionality and capacity meet your requirements.

  b. When you obtain an activation file, make sure the provided license key and hardware information are correct.

  c. Install the activation file on the correct target device.

- Make sure no one else is performing license management tasks while you are managing licenses on the device.
- You can manage licenses only on the default context. Any license operation performed on the default context takes effect on all contexts. For information about contexts, see *Virtual Technologies Configuration Guide.*

File operation restrictions

When you manage DID files, activation files, or Uninstall files, follow these restrictions and guidelines:

- To avoid licensing error, do not modify the name of a DID file, activation file, or Uninstall file, or edit the file content.
- Before you install an activation file, download the activation file to the storage media of the device such as flash memory. When installing an activation file, the device automatically copies the activation file to the **license** folder in the root directory of the storage media. The **license** folder stores important files for licensing. For licensed features to function correctly, do not delete or modify the **license** folder or the files in this folder.

License consolidation

License consolidation combines multiple licenses to create one activation file. It delivers the following benefits:

- **Ease of license installation and management**—This feature enables you to install one activation file to activate multiple licenses, without having to install one activation file for each of them.
- **Storage conservation**—This feature enables the device to store one activation file for multiple licenses, which conserves the license storage space on the device. Licenses that support consolidation are called consolidable licenses.

Configuring local licensing

About local licensing

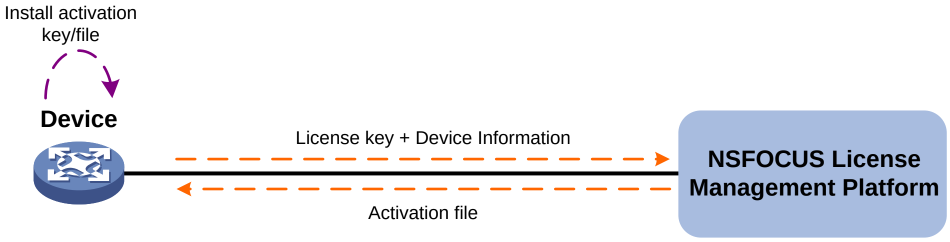
Local licensing requires license activation device by device. It is applicable to small-sized networks.

To install a license on a device:

1. Obtain the license key and the required device information.
2. Contact NSFOCUS sales representatives or technical support to obtain an activation file based on the license key and the device information.
3. Install the activation file on the device to activate the license.

The activation file for a license is device locked. You cannot install the activation file for one device to activate the license on another device.

Figure 1 Local licensing procedure


Install activation key/file
Device
License key + Device Information
Activation file
NSFOCUS License Management Platform

Installing a license

The procedures for installing formal licenses and trial licenses are the same .

Identifying the license storage

To identify the free space of the license storage, execute the following command in any view:

```
display license feature
```

From the command output, view the **Total** and **Usage** fields to examine whether the remaining license storage is sufficient for installing new licenses. If the remaining license storage is not sufficient, compress the license storage.

Compressing the license storage

About this task

The license storage stores licensing information and has a fixed size.

You can compress the license storage to delete expired and uninstalled license information to ensure sufficient storage space for installing new licenses.

If no license has been installed on the device, you do not need to compress the license storage.

Prerequisites

Back up the Uninstall keys or Uninstall files for the uninstalled licenses for subsequent license transfer or license uninstallation.

If uninstalled licenses or expired licenses exist on the device, the compression operation will make the DID change. You will be unable to install the activation file obtained by using the old DID on the device. As a best practice, install all activation files registered with the old DID before performing a compression.

If you have not installed an activation file registered with the old DID, take the following actions:

- If the license storage is sufficient, install the activation file on the device. For more information, see the licensing guide for the device.
- If the license storage is insufficient and the activation file cannot be installed after the compression, contact NSFOCUS Support.

Procedure

1. Enter system view.

   `system-view`

2. Compress the license storage.

   `license compress slot` *`slot-number`*

Obtaining required information for license registration

To obtain SN and DID information, execute the following command in any view:

`display license device-id slot` *`slot-number`*

Installing an activation file

About this task

△ **CAUTION:**

Back up an activation file before you install it. If the activation file is inadvertently deleted or becomes unavailable for some other reason, you can use the backup activation file to restore the license.

To obtain a license, install an activation file for the license on the device.

Prerequisites

Use FTP or TFTP to upload the activation file to be installed to the device. If FTP is used to transfer the activation file, set it in binary mode.

Installing an activation file

1. Enter system view.

   `system-view`

2. Install an activation file.

   `license activation-file install` *`license-file`* **`slot`** *`slot-number`*

   You can install a single .ak file or multiple .ak files through one operation. To install multiple .ak files, save all activation files in the same directory and specify the directory as the value of the *`license-file`* argument

Display and maintenance commands for license management

Execute **`display`** commands in any view.

Task	Command
Display detailed license information.	**display license** [**activation-file**] [**slot** *slot-number*]
Display the SN and DID information.	**display license device-id slot** *slot-number*
Display brief feature license information.	**display license feature**

Contents

Managing the device

About device management

This chapter describes how to configure basic device parameters and manage the device.

Device management tasks at a glance

All device management tasks are optional. You can perform any of the tasks in any order.

- Configuring basic parameters
  - Configuring the device name
  - Configuring the system ID
  - Configuring the system time
  - Enabling displaying the copyright statement
  - Configuring banners
- Configuring security parameters
  - Disabling password recovery capability
  - Disabling USB interfaces
- Adjusting device capacities
  - Setting the port status detection timer
- Monitoring the device
  - Monitoring CPU usage
  - Monitoring CPU core usage
  - Setting memory alarm thresholds
  - Configuring resource monitoring
  - Setting the temperature alarm thresholds
- Managing resources
  - Verifying and diagnosing transceiver modules
- Maintaining the device
  - Scheduling a task
  - Locating devices
  - Rebooting the device
  - Restoring the factory-default configuration

Configuring the device name

About this task

A device name (also called hostname) identifies a device in a network and is used in CLI view prompts. For example, if the device name is **Sysname**, the user view prompt is <Sysname>.

Procedure

1. Enter system view.

```
system-view
```

2. Configure the device name.

```
sysname sysname
```

By default, the device name is NSFOCUS.

Configuring the system ID

About this task

You can use the system ID to indicate the position or functionality of the device or any other information.

Procedure

1. Enter system view.

```
system-view
```

2. Configure the system ID.

```
sysid system-id
```

By default, the device does not have a system ID.

Configuring the system time

About the system time

Correct system time is essential to network management and communication. Configure the system time correctly before you run the device on the network.

The device can use one of the following methods to obtain the system time:

- Uses the locally set system time, and then uses the clock signals generated by its built-in crystal oscillator to maintain the system time.
- Periodically obtains the UTC time from an NTP source, and uses the UTC time, time zone, and daylight saving time to calculate the system time. For more information about NTP, see *Network Management and Monitoring Configuration Guide*.

The system time calculated by using the UTC time from a time source is more precise.

Restrictions and guidelines for configuring the system time

After you execute the **clock protocol none** command, the **clock datetime** command determines the system time, whether or not the time zone or daylight saving time has been configured.

If you configure or change the time zone or daylight saving time after the device obtains the system time, the device recalculates the system time. To view the system time, execute the **display clock** command.

This feature is supported only on the default context. All contexts on the device use the same system time.

System time configuration tasks at a glance

To configure the system time, perform the following tasks:

1. Configuring the system time

Choose one of the following tasks:

- Setting the system time at the CLI
- Obtaining the UTC time through a time protocol

2. (Optional.) Setting the time zone

   Make sure each network device uses the time zone of the place where the device resides.

3. (Optional.) Setting the daylight saving time

   Make sure each network device uses the daylight saving time parameters of the place where the device resides.

Setting the system time at the CLI

1. Enter system view.

   `system-view`

2. Configure the device to use the local system time.

   `clock protocol none`

   By default, the device uses the NTP time source specified on the default context.

   If you execute the **`clock protocol`** command multiple times, the most recent configuration takes effect.

3. Return to user view.

   `quit`

4. Set the local system time.

   `clock datetime` *`time date`*

   By default, the system time is UTC time 00:00:00 01/01/2011.

△ **CAUTION:**

This command changes the system time, which affects the execution of system time-related features (for example, scheduled tasks) and collaborative operations of the device with other devices (for example, log reporting and statistics collection). Before executing this command, make sure you fully understand its impact on your live network.

Obtaining the UTC time through a time protocol

Restrictions and guidelines

If the NTP signals are lost, the device uses the clock signals generated by its built-in crystal oscillator to maintain the system time. After the NTP signals recover, the device obtains the UTC time again through NTP.

Procedure

1. Enter system view.

   `system-view`

2. Specify the protocol for obtaining the UTC time.

   `clock protocol ntp context` *`context-id`*

   By default, the device uses the NTP time source specified on the default context.

   If you execute the **`clock protocol`** command multiple times, the most recent configuration takes effect.

   Support for the **`context`** *`context-id`* option depends on the device model. For more information, see device management in *Fundamentals Command Reference*.

3. Configure time protocol parameters.

   For more information about NTP configuration, see *Network Management and Monitoring Configuration Guide*.

Setting the time zone

1. Enter system view.

   `system-view`

2. Set the time zone.

   `clock timezone` *`zone-name`* { **`add`** | **`minus`** } *`zone-offset`*

   By default, the system uses the UTC time zone.

Setting the daylight saving time

1. Enter system view.

   `system-view`

2. Set the daylight saving time.

   `clock summer-time` *`name start-time start-date end-time end-date add-time`*

   By default, the daylight saving time is not set.

Enabling displaying the copyright statement

About this task

This feature enables the device to display the copyright statement in the following situations:

- When a Telnet or SSH user logs in.
- When a console user quits user view. This is because the device automatically tries to restart the user session.

If you disable displaying the copyright statement, the device does not display the copyright statement in any situations.

Procedure

1. Enter system view.

   `system-view`

2. Enable displaying the copyright statement.

   `copyright-info enable`

   By default, displaying the copyright statement is enabled.

Configuring banners

About this task

Banners are messages that the system displays when a user logs in.

The system supports the following banners:

- **Legal banner**—Appears after the copyright statement. To continue login, the user must enter **Y** or press **Enter**. To quit the process, the user must enter **N**. **Y** and **N** are case insensitive.

- **Message of the Day (MOTD) banner**—Appears after the legal banner and before the login banner.
- **Login banner**—Appears only when password or scheme authentication is configured.
- **Shell banner**—Appears before the user enters user view.

The system displays the banners in the following order: legal banner, MOTD banner, login banner, and shell banner.

Banner input methods

You can configure a banner by using one of the following methods:

- Input the entire command line in a single line.

  The banner cannot contain carriage returns. The entire command line, including the command keywords, the banner, and the delimiters, can have a maximum of 511 characters. The delimiters for the banner can be any printable character but must be the same. You cannot press **Enter** before you input the end delimiter.

  For example, you can configure the shell banner "Have a nice day." as follows:

  ```
  <System> system-view
  [System] header shell %Have a nice day.%
  ```

- Input the command line in multiple lines.

  The banner can contain carriage returns. A carriage return is counted as two characters.

  To input a banner configuration command line in multiple lines, use one of the following methods:

  - Press **Enter** after the final command keyword, type the banner, and end the final line with the delimiter character %. The banner plus the delimiter can have a maximum of 1999 characters.

    For example, you can configure the banner "Have a nice day." as follows:

    ```
    <System> system-view
    [System] header shell
    Please input banner content, and quit with the character '%'.
    Have a nice day.%
    ```

  - After you type the final command keyword, type any printable character as the start delimiter for the banner and press **Enter**. Then, type the banner and end the final line with the same delimiter. The banner plus the end delimiter can have a maximum of 1999 characters.

    For example, you can configure the banner "Have a nice day." as follows:

    ```
    <System> system-view
    [System] header shell A
    Please input banner content, and quit with the character 'A'.
    Have a nice day.A
    ```

  - After you type the final command keyword, type the start delimiter and part of the banner. Make sure the final character of the final string is different from the start delimiter. Then, press **Enter**, type the rest of the banner, and end the final line with the same delimiter. The banner plus the start and end delimiters can have a maximum of 2002 characters.

    For example, you can configure the banner "Have a nice day." as follows:

    ```
    <System> system-view
    [System] header shell AHave a nice day.
    Please input banner content, and quit with the character 'A'.
    A
    ```

Procedure

1. Enter system view.

```
system-view
```

2. Configure the legal banner.

   header legal *text*

3. Configure the MOTD banner.

   header motd *text*

4. Configure the login banner.

   header login *text*

5. Configure the shell banner.

   header shell *text*

Disabling password recovery capability

About this task

Password recovery capability controls console user access to the device configuration and SDRAM from BootWare menus. For more information about BootWare menus, see the release notes.

If password recovery capability is enabled, a console user can access the device configuration without authentication to configure a new password.

If password recovery capability is disabled, console users must restore the factory-default configuration before they can configure new passwords. Restoring the factory-default configuration deletes the next-startup configuration files.

To enhance system security, disable password recovery capability.

Restrictions and guidelines

This feature is supported only on the default context.

To access the device configuration without authentication, you must connect to the master device and access the BootWare menu while the master device is starting up.

Procedure

1. Enter system view.

   system-view

2. Disable password recovery capability.

   undo password-recovery enable

   By default, password recovery capability is enabled.

Disabling USB interfaces

About this task

You can upload or download files or use 3G or 4G capabilities on the device through USB interfaces. By default, all USB interfaces are enabled. You can disable USB interfaces as needed. If USB interfaces are disabled, file upload and download and 3G/4G capabilities are not available.

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

1. Enter system view.

   system-view

2. Disable USB interfaces.

   `usb disable`

   By default, all USB interfaces are enabled.

Setting the port status detection timer

About this task

The device starts a port status detection timer when a port is shut down by a protocol. Once the timer expires, the device brings up the port so the port status reflects the port's physical status.

Procedure

1. Enter system view.

   `system-view`

2. Set the port status detection timer.

   `shutdown-interval` *`time`*

   The default setting is 30 seconds.

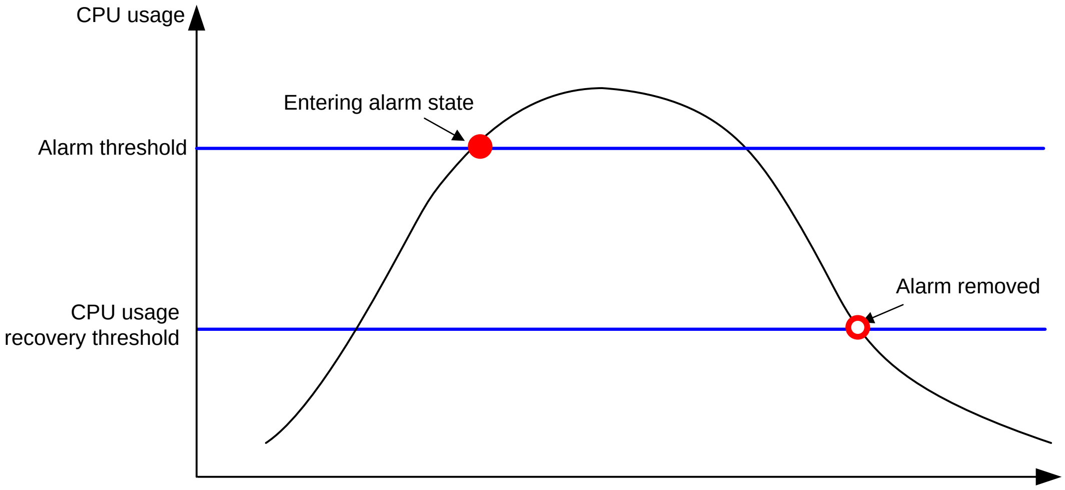
Monitoring CPU usage

About this task

To monitor CPU usage, the device performs the following operations:

- Samples CPU usage at 1-minute intervals and compares the samples with the CPU usage threshold and the CPU usage recovery threshold.
  - If a sample is greater than or equal to the CPU usage threshold, the device determines the CPU usage is high and sends traps to affected service modules and processes.
  - If a sample decreases to or below the CPU usage recovery threshold, the device determines the CPU usage has recovered and sends traps to affected service modules and processes.
- Samples and saves CPU usage at a configurable interval if CPU usage tracking is enabled. You can use the **`display cpu-usage history`** command to display the historical CPU usage statistics in a coordinate system.

Figure 1 CPU alarms and alarm-removed notifications



Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the CPU usage alarm thresholds.

   ```
   monitor cpu-usage threshold severe-threshold recovery-threshold
   recovery-threshold [ slot slot-number [ cpu cpu-number ] ]
   ```

   By default, the CPU usage alarm threshold is 70%, and the CPU usage recovery threshold is 30%.

   CAUTION:

   If you set the severe CPU usage alarm threshold to a too low value, the device will reach the threshold easily. Normal services will be affected.

3. Set the sampling interval for CPU usage tracking.

   ```
   monitor cpu-usage interval interval [ slot slot-number [ cpu
   cpu-number ] ]
   ```

   By default, the sampling interval for CPU usage tracking is 1 minute.

4. Enable CPU usage tracking.

   ```
   monitor cpu-usage enable [ slot slot-number [ cpu cpu-number ] ]
   ```

   By default, CPU usage tracking is enabled.

5. Enable periodic CPU usage logging.

   ```
   monitor cpu-usage logging slot slot-number cpu cpu-number interval
   interval-time
   ```

   By default, periodic CPU usage logging is disabled.

Monitoring CPU core usage

About this task

The device samples CPU core usage at 5-second intervals and calculates the average value during each CPU core usage statistics interval. If the value during an interval is greater than the CPU core usage threshold, the device issues an alarm and logs the event.

Restrictions and guidelines

As a best practice, set this argument to a multiple of the sampling interval, which is fixed at 5 seconds. If you do not do so, the actual statistics interval is the biggest multiple of the sampling interval that is smaller than the setting. For example, if you set this argument to 12 seconds, the actual statistics interval is 10 seconds.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set CPU core usage statistics intervals.

   ```
   monitor cpu-usage statistics-interval interval slot slot-number cpu
   cpu-number core core-id-list
   ```

   By default, the CPU core usage statistics interval is 60 seconds.

3. Set CPU core alarm resending intervals.

   ```
   monitor resend cpu-usage core-interval core-interval [ slot
   slot-number [ cpu cpu-number ] ]
   ```

By default, the CPU core alarm resending interval is 300 seconds.

Setting memory alarm thresholds

About memory alarm thresholds

To ensure correct operation and improve memory efficiency, the system performs the following operations:

- Samples memory usage at 1-minute intervals. If the sample is equal to or greater than the memory usage threshold, the device sends a trap.
- Monitors the amount of free memory space in real time. If the amount of free memory space reaches the minor, severe, or critical alarm threshold, the system issues an alarm to affected service modules and processes.

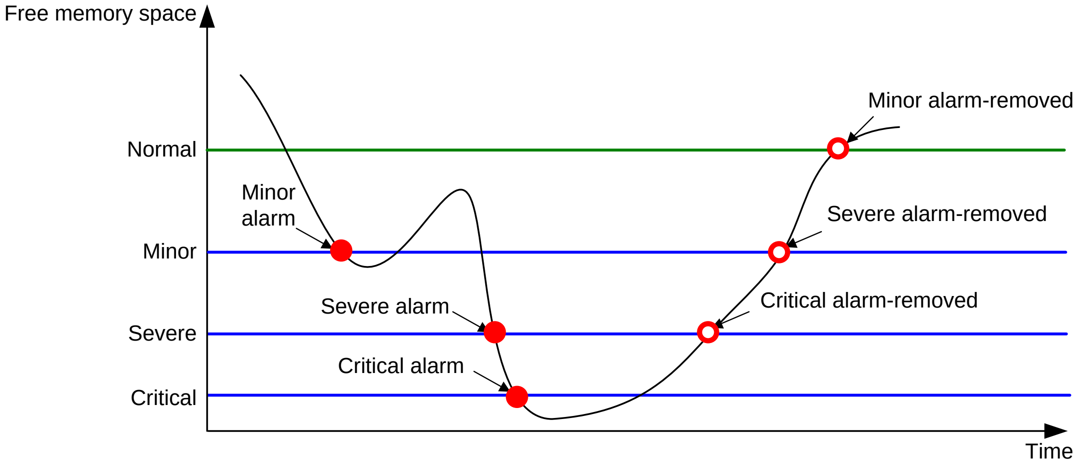
As shown in Table 1 and Figure 2, the system supports the following free-memory thresholds:

- Normal state threshold.
- Minor alarm threshold.
- Severe alarm threshold.
- Critical alarm threshold.

Table 1 Memory alarm notifications and memory alarm-removed notifications

Notification	Triggering condition	Remarks
Minor alarm notification	The amount of free memory space decreases below the minor alarm threshold.	After generating and sending a minor alarm notification, the system does not generate and send any additional minor alarm notifications until the minor alarm is removed.
Severe alarm notification	The amount of free memory space decreases below the severe alarm threshold.	After generating and sending a severe alarm notification, the system does not generate and send any additional severe alarm notifications until the severe alarm is removed.
Critical alarm notification	The amount of free memory space decreases below the critical alarm threshold.	After generating and sending a critical alarm notification, the system does not generate and send any additional critical alarm notifications until the critical alarm is removed.
Critical alarm-removed notification	The amount of free memory space increases above the severe alarm threshold.	N/A
Severe alarm-removed notification	The amount of free memory space increases above the minor alarm threshold.	N/A
Minor alarm-removed notification	The amount of free memory space increases above the normal state threshold.	N/A

Figure 2 Memory alarm notifications and alarm-removed notifications



Restrictions and guidelines

Restrictions and guidelines

This feature is supported only on the default context.

If a memory alarm occurs, delete unused configuration items or disable some features to increase the free memory space. Because the memory space is insufficient, some configuration items might not be able to be deleted.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the memory usage threshold.

   ```
   memory-threshold [ slot slot-number [ cpu cpu-number ] ] usage memory-threshold
   ```

   By default, the memory usage threshold is 95%.

3. Set the free-memory thresholds.

   ```
   memory-threshold [ slot slot-number [ cpu cpu-number ] ] [ ratio ] minor minor-value severe severe-value critical critical-value normal normal-value
   ```

   The default settings vary by device model. To view the default settings, use the **undo memory-threshold** command to restore the default settings and then execute the **display memory-threshold** command.

4. Enable periodic memory usage logging.

   ```
   monitor memory-usage logging slot slot-number cpu cpu-number interval interval-time
   ```

   By default, periodic memory usage logging is disabled.

Configuring resource monitoring

Monitoring the total inbound bandwidth usage

About this task

If the total inbound traffic remains greater than or equal to the total inbound bandwidth usage threshold for the specified duration, the device sends an alarm. If the alarm state persists, the device resends the alarm at 5-second intervals.

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

1. Enter system view.

   system-view

2. Set the total inbound bandwidth usage threshold.

   monitor resource-usage bandwidth inbound threshold *threshold-value* [**duration** *duration-value*]

   By default, the total inbound bandwidth usage threshold is not set. The bandwidth usage alarm feature is disabled.

Monitoring the aggregate interface usage

About this task

When the number of created Layer 2 or Layer 3 aggregate interfaces reaches the aggregate interface usage threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 3-hour intervals.

Procedure

1. Enter system view.

   system-view

2. Set aggregate interface usage thresholds.

   monitor resource-usage { **bridge-aggregation** | **route-aggregation** } **threshold** *threshold-value*

   By default, no aggregate interface usage thresholds are set. The aggregate interface usage alarm feature is disabled.

Monitoring the inner interface throughput

About this task

When the inner interface throughput reaches the threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 10-minute intervals.

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

1. Enter system view.

   system-view

2. Set the inner interface throughput threshold.

   monitor resource-usage blade-throughput threshold *threshold-value*

   By default, the inner interface throughput threshold is not set. The inner interface throughput alarm feature is disabled.

Monitoring the number of contexts

About this task

When the total number of contexts created on the device reaches the global context usage threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 6-hour intervals.

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

1. Enter system view.

   system-view

2. Set the global context usage threshold.

   monitor resource-usage context threshold *threshold-value*

   By default, the global context usage threshold is not set. The global context usage alarm feature is disabled.

Monitoring the number of NAT mappings

About this task

When the number of NAT mappings reaches the NAT mapping threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 3-hour intervals.

In the current software version, this feature counts only static NAT mappings and effective NAT server mappings. To display the status of NAT server mappings, execute the **display nat server** command.

Procedure

1. Enter system view.

   system-view

2. Set the NAT mapping threshold.

   monitor resource-usage nat threshold *threshold-value*

   By default, the NAT mapping threshold is not set. The NAT mapping alarm feature is disabled.

Monitoring the number of security policy rules

About this task

When the number of created security policy rules reaches the security policy rule usage threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 6-hour intervals.

Procedure

1. Enter system view.

   system-view

2. Set security policy rule usage thresholds.

   monitor resource-usage security-policy { **ip** | **ipv6** } **threshold** *threshold-value*

   By default, no security policy rule thresholds are set. The security policy rule alarm feature is disabled.

Monitoring the number of sessions

About this task

When the number of sessions reaches the session usage threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 10-minute intervals.

Procedure

1. Enter system view.

   system-view

2. Set the session usage threshold.

   monitor resource-usage session-count [**slot** *slot-number*] **threshold** *threshold-value*

   By default, no session usage thresholds are set. The session usage alarm feature is disabled.

Monitoring the session establishment rate

About this task

When the session establishment rate reaches the threshold, the device sends an alarm. If the alarm state persists, the device resends the alarm at 10-minute intervals.

Procedure

1. Enter system view.

   system-view

2. Set the session establishment rate threshold.

   monitor resource-usage session-rate [**slot** *slot-number*] **threshold** *threshold-value*

   By default, no session establishment rate thresholds are set. The session establishment rate alarm feature is disabled.

Setting the temperature alarm thresholds

About this task

The device monitors its temperature based on the following thresholds:

- Low-temperature threshold.
- High-temperature warning threshold.
- High-temperature alarming threshold.

When the device temperature drops below the low-temperature threshold or reaches the high-temperature warning or alarming threshold, the device performs the following operations:

- Sends log messages and traps.
- Sets LEDs on the device panel.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

1. Enter system view.

   `system-view`

2. Configure the temperature alarm thresholds.

   `temperature-limit slot` *`slot-number`* { **`hotspot`** | **`inflow`** | **`outflow`** } *`sensor-number lowlimit warninglimit`* [*`alarmlimit`*]

   The defaults vary by temperature sensor model. To view the defaults, execute the **`undo temperature-limit`** and **`display environment`** commands in turn.

   The high-temperature alarming threshold must be higher than the high-temperature warning threshold, and the high-temperature warning threshold must be higher than the low-temperature threshold.

   Support for the command depends on the device model. For more information, see the command reference.

Verifying and diagnosing transceiver modules

Verifying transceiver modules

About this task

You can use one of the following methods to verify the genuineness of a transceiver module:

- Display the key parameters of a transceiver module, including its transceiver type, connector type, central wavelength of the transmit laser, transfer distance, and vendor name.
- Display its electronic label. The electronic label is a profile of the transceiver module and contains the permanent configuration, including the serial number, manufacturing date, and vendor name. The data was written to the transceiver module or the device's storage component during debugging or testing of the transceiver module or device.

The device regularly checks transceiver modules for their vendor names. If a transceiver module does not have a vendor name or the vendor name is not NSFOCUS, the device repeatedly outputs traps and log messages. For information about logging rules, see information center in *Network Management and Monitoring Configuration Guide*.

Procedure

To verify transceiver modules, execute the following commands in any view:

- Display the key parameters of transceiver modules.

  `display transceiver interface` [*`interface-type interface-number`*]

- Display the electrical label information of transceiver modules.

display transceiver manuinfo { **controller** [*controller-type controller-number*] | **interface** [*interface-type interface-number*] }

Diagnosing transceiver modules

About this task

The device provides the alarm and digital diagnosis functions for transceiver modules. When a transceiver module fails or is not operating correctly, you can perform the following tasks:

- Check the alarms that exist on the transceiver module to identify the fault source.
- Examine the key parameters monitored by the digital diagnosis function, including the temperature, voltage, laser bias current, TX power, and RX power.

Procedure

To diagnose transceiver modules, execute the following commands in any view:

- Display transceiver alarms.

  display transceiver alarm { **controller** [*controller-type controller-number*] | **interface** [*interface-type interface-number*] }
- Display the current values of the digital diagnosis parameters on transceiver modules.

  display transceiver diagnosis { **controller** [*controller-type controller-number*] | **interface** [*interface-type interface-number*] }

Support for the **controller** keyword depends on the device model. For more information, see device management commands in *Fundamentals Command Reference*.

Scheduling a task

About task scheduling

You can schedule the device to automatically execute a command or a set of commands without administrative interference.

You can configure a periodic schedule or a non-periodic schedule. A non-periodic schedule is not saved to the configuration file and is lost when the device reboots. A periodic schedule is saved to the startup configuration file and is automatically executed periodically.

Restrictions and guidelines

- To assign a command (command A) to a job, you must first assign the job the command or commands for entering the view of command A.
- Make sure all commands in a schedule are compliant to the command syntax. The system does not check the syntax when you assign a command to a job.
- A schedule cannot contain any one of these commands: **telnet**, **ftp**, **ssh2**, and **monitor process**.
- A schedule does not support user interaction. If a command requires a yes or no answer, the system always assumes that a **Y** or **Yes** is entered. If a command requires a character string input, the system assumes that either the default character string (if any) or a null string is entered.
- A schedule is executed in the background, and no output (except for logs, traps, and debug information) is displayed for the schedule.

Procedure

1. Enter system view.

   system-view

2. Create a job.

   scheduler job *job-name*

3. Assign a command to the job.

   command *id command*

   By default, no command is assigned to a job.

   You can assign multiple commands to a job. A command with a smaller ID is executed first.

4. Exit to system view.

   quit

5. Create a schedule.

   scheduler schedule *schedule-name*

6. Assign a job to the schedule.

   job *job-name*

   By default, no job is assigned to a schedule.

   You can assign multiple jobs to a schedule. The jobs will be executed concurrently.

7. Assign user roles to the schedule.

   user-role *role-name*

   By default, a schedule has the user role of the schedule creator.

   You can assign a maximum of 64 user roles to a schedule. A command in a schedule can be executed if it is permitted by one or more user roles of the schedule.

8. Specify the execution time for the schedule.

   Choose one option as needed:

   - Execute the schedule at specific points of time.

     time at *time date*

     time once at *time* [**month-date** *month-day* | **week-day** *week-day*&<1-7>]

   - Execute the schedule after a period of time.

     time once delay *time*

   - Execute the schedule at the specified time on every specified day in a month or week.

     time repeating at *time* [**month-date** [*month-day* | **last**] | **week-day** *week-day*&<1-7>]

   - Execute the schedule periodically from the specified time on.

     time repeating [**at** *time* [*date*]] **interval** *interval*

   By default, no execution time is specified for a schedule.

   The **time** commands overwrite each other. The most recently executed command takes effect.

9. (Optional.) Set the size of the job execution log file.

   scheduler logfile size *value*

   By default, the size of the job execution log file is 16 KB.

   The job execution log file stores the execution information of jobs. If the file is full, old records are deleted to make room for new records. If the size of the log information to be written to the file is greater than the file size, the excessive information is not written to the file.

Locating devices

About device locating

The device provides SYS LEDs for device locating. The **locator blink** *blink-time* command flashes the LEDs quickly for a specified period of time unless you execute the **locator blink stop** command.

Restrictions and guidelines

This feature is supported only on the default context.

Starting LED flashing

To start LED flashing, execute one of the following commands in user view:

locator [**slot** *slot-number*] **blink** *blink-time*

Stopping LED flashing

To stop LED flashing, execute one of the following commands in user view:

locator [**slot** *slot-number*] **blink stop**

Rebooting the device

About device reboot

The following device reboot methods are available:

- Schedule a reboot at the CLI, so the device automatically reboots at the specified time or after the specified period of time.
- Immediately reboot the device at the CLI.

  During the reboot process, the device performs the following operations:

  a. Resets all of its chips.

  b. Uses the BootWare to verify the startup software package, decompress the package, and load the images.

  c. Initializes the system.
- Power off and then power on the device. This method might cause data loss, and is the least-preferred method.

Using the CLI, you can reboot the device from a remote host.

Restrictions and guidelines for device reboot

For data security, the device does not reboot while it is performing file operations.

Rebooting devices immediately at the CLI

Prerequisites

Perform the following steps in any view:

1. Verify that the next-startup configuration file is correctly specified.

   `display startup`

   For more information about the **`display startup`** command, see *Fundamentals Command Reference*.

2. Verify that the startup image files are correctly specified.

   `display boot-loader`

   If one main startup image file is damaged or does not exist, you must specify another main startup image file before rebooting the device.

   For more information about the **`display boot-loader`** command, see *Fundamentals Command Reference*.

3. Save the running configuration to the next-startup configuration file.

   `save`

   To avoid configuration loss, save the running configuration before a reboot.

   For more information about the **`save`** command, see *Fundamentals Command Reference*.

Procedure

To reboot the device immediately at the CLI, execute one of the following commands in user view:

`reboot` `[` **`slot`** *`slot-number`* `] [` **`force`** `]`

CAUTION:

- A device reboot might result in service interruption. Before using this command, make sure you fully understand its impact on your live network.
- Use the **`force`** keyword to reboot the device only when the system is faulty or fails to start up normally. A forced device reboots might cause file system damage. Before using the **`force`** keyword to reboot the device, make sure you understand its impact.

Scheduling a device reboot

Restrictions and guidelines

The automatic reboot configuration takes effect on all member devices. It will be canceled if a master/subordinate switchover occurs.

The device supports only one device reboot schedule. If you execute the **`scheduler reboot`** command multiple times, the most recent configuration takes effect.

Procedure

To schedule a reboot, execute one of the following commands in user view:

- **`scheduler reboot at`** *`time`* `[` *`date`* `]`
- **`scheduler reboot delay`** *`time`*

By default, no device reboot time is specified.

CAUTION:

This command enables the device to reboot at a scheduled time, which causes service interruption.

Before using this command, make sure you fully understand its impact on your live network.

Restoring the factory-default configuration

About this task

If you want to use the device in a different scenario or you cannot troubleshoot the device by using other methods, use this task to restore the factory-default configuration.

This task does not delete **.bin** files and license files.

Restrictions and guidelines

This feature is supported only on the default context.

Procedure

To restore the factory-default configuration for the device, execute the following command in user view:

`restore factory-default`

△ **CAUTION:**

This command restores the device to the factory default settings. Before using this command, make sure you fully understand its impact on your live network.

Display and maintenance commands for device management configuration

ⓘ **IMPORTANT:**

- Support for the **`display alarm`**, **`display environment`**, **`display fan`**, and **`display power`** commands depends on the device model.
- Support for the **`cf-card`**, **`harddisk`**, and **`usb`** keywords in the **`display device`** command depends on the device model.
- Support for the **`context`** keyword in the **`display diagnostic-information`** command depends on the device model.
- Support for the *`fan-id`* argument in the **`display fan`** command depends on the device model.
- Support for the *`power-id`* argument in the **`display power`** command depends on the device model.
- Support for the **`context`** keyword in the **`display system stable state`** command depends on the device model.
- Support for the **`controller`** keyword in the following commands depends on the device model: **`display transceiver alarm`**, **`display transceiver diagnosis`**, and **`display transceiver manuinfo`**.

For more information, see device management commands in *Fundamentals Command Reference*.

Execute **`display`** commands in any view. Execute the **`reset scheduler logfile`** command in user view. Execute the **`reset version-update-record`** command in system view.

Task	Command
Display device alarm information.	**`display alarm`** [**`slot`** *`slot-number`*]

Display the system time, date, time zone, and daylight saving time.	**display clock**
Display the copyright statement.	**display copyright**
Display CPU usage statistics.	**display cpu-usage** [**summary**] [**slot** *slot-number* [**cpu** *cpu-number* [**core** { *core-number* | **all** }]]] **display cpu-usage** [**control-plane** | **data-plane**] [**summary**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display CPU usage monitoring settings.	**display cpu-usage configuration** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display the historical CPU usage statistics in a coordinate system.	**display cpu-usage history** [**job** *job-id*] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display hardware information.	**display device** [**harddisk** | **usb**] [**slot** *slot-number* [**subslot** *subslot-number*] | **verbose**]
Display electronic label information for the device.	**display device manuinfo** [**slot** *slot-number*]
Display or save operating information for features and hardware modules.	**display diagnostic-information** [**hardware** | **infrastructure** | **l2** | **l3** | **service**] [**key-info**] [*filename*]
Display device temperature information.	**display environment** [**slot** *slot-number*]
Display the operating states of fan trays.	**display fan** [**slot** *slot-number* [*fan-id*]]
Display memory usage statistics.	**display memory** [**summary**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display memory alarm thresholds and statistics.	**display memory-threshold** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display power supply information.	**display power** [**slot** *slot-number* [*power-id*]]
Display job configuration information.	**display scheduler job** [*job-name*]
Display job execution log information.	**display scheduler logfile**
Display the automatic reboot schedule.	**display scheduler reboot**
Display schedule information.	**display scheduler schedule** [*schedule-name*]
Display system stability and status information.	**display system stable state**
Display system version information.	**display version**
Display startup software image upgrade records.	**display version-update-record**
Clear job execution log information.	**reset scheduler logfile**
Clear startup software image upgrade records.	**reset version-update-record**

NOTE:

- Executing the **display cpu-usage**, **display cpu-usage configuration**, **display cpu-usage history**, or **display memory** command on a context displays information for the context.
- The **display device** command displays device information about the physical devices, whether you execute the command on the default context or on a non-default context.

NOTE:

The following commands are supported only on the default context:

- **display alarm**
- **display device manuinfo**
- **display environment**
- **display fan**
- **display memory-threshold**
- **display power**
- **display version-update-record**
- **reset scheduler logfile**
- **reset version-update-record**

Contents

Configuring FTP

About FTP

File Transfer Protocol (FTP) is an application layer protocol for transferring files from one host to another over an IP network. It uses TCP port 20 to transfer data and TCP port 21 to transfer control commands.

FTP is based on the client/server model. The device can act as the FTP server or FTP client.

FTP file transfer modes

FTP supports the following transfer modes:

- **Binary mode**—Used to non-text files, such as **.app**, **.bin**, and **.btm** files.
- **ASCII mode**—Used to transfer text files, such as **.txt**, **.bat**, and **.cfg** files.

When the device acts as the FTP client, you can set the transfer mode (**binary** by default). When the device acts as the FTP server, the transfer mode is determined by the FTP client.

FTP operation modes

FTP can operate in either of the following modes:

- **Active mode (PORT)**—The FTP server initiates the TCP connection. This mode is not suitable when the FTP client is behind a firewall, for example, when the FTP client resides in a private network.
- **Passive mode (PASV)**—The FTP client initiates the TCP connection. This mode is not suitable when the server does not allow the client to use a random unprivileged port greater than 1024.

FTP operation mode varies depending on the FTP client program.

Using the device as an FTP server

To use the device as an FTP server, you must enable the FTP server and configure authentication and authorization on the device. Other commands are optional.

FTP server configuration tasks at a glance

To use the device as an FTP server, perform the following tasks:

1. Enabling the FTP server
2. Configuring client authentication and authorization
3. (Optional.) Configuring FTP server access control
4. (Optional.) Setting connection management parameters
5. (Optional.) Specifying an SSL server policy for SFTP connections
6. (Optional.) Setting the DSCP value for outgoing FTP packets
7. (Optional.) Manually releasing FTP connections

Enabling the FTP server

1. Enter system view.

   ```
   system-view
   ```

2. Enable the FTP server.

   ```
   ftp server enable
   ```

   By default, the FTP server is disabled.

Configuring client authentication and authorization

Perform this task on the FTP server to authenticate FTP clients and set the authorized directories that authenticated clients can access.

The following authentication modes are available:

- **Local authentication**—The device looks up the client's username and password in the local user account database. If a match is found, authentication succeeds.
- **Remote authentication**—The device sends the client's username and password to a remote authentication server for authentication. The user account is configured on the remote authentication server rather than the device.

The following authorization modes are available:

- **Local authorization**—The device assigns authorized directories to FTP clients based on the locally configured authorization attributes.
- **Remote authorization—**A remote authorization server assigns authorized directories on the device to FTP clients.

For more information about configuring authentication and authorization, see AAA in *Security Configuration Guide*.

Configuring FTP server access control

About this task

Use ACLs to prevent unauthorized access. If an applied ACL does not exist or does not have any rules, no user login restriction is applied. If the ACL exists and has rules, only FTP clients permitted by the ACL can access the device.

Restrictions and guidelines

When no ACL is applied, all FTP clients can access the FTP server. To control FTP access, specify an ACL that exists and has rules so only FTP clients permitted by the ACL can access the FTP server. If you specify an ACL that does not exist or does not have rules, no FTP clients can access the FTP server.

If a VPN instance is specified in an ACL rule, the ACL rule applies only to the packets of the VPN instance. If no VPN instance is specified in an ACL rule, the ACL rule applies only to the packets on the public network.

The ACL takes effect only for FTP connections to be established. It does not impact existing FTP connections.

For more information about ACL, see *ACL and QoS Configuration Guide.*

If you configure FTP server access control multiple times, the most recent configuration takes effect.

Procedure

1. Enter system view.

system-view

2. (Optional.) Use an ACL to control access to the FTP server.

   IPv4:

   ftp server acl { *advanced-acl-number* | *basic-acl-number* | *mac mac-acl-number* }

   IPv6:

   ftp server acl ipv6 { *advanced-acl-number* | *basic-acl-number* | **mac** *mac-acl-number* }

   By default, no ACL is used for access control.

3. (Optional.) Enable logging for FTP login attempts that are denied by the FTP login control ACL.

   ftp server acl-deny-log enable

   By default, logging is disabled for FTP login attempts that are denied by the FTP login control ACL.

Setting connection management parameters

1. Enter system view.

   system-view

2. Set the FTP connection idle-timeout timer.

   ftp timeout *minutes*

   By default, the FTP connection idle-timeout timer is 30 minutes.

   If no data transfer occurs on an FTP connection before the idle-timeout timer expires, the FTP server closes the FTP connection.

3. Set the maximum number of concurrent FTP users.

   aaa session-limit ftp *max-sessions*

   By default, the maximum number of concurrent FTP users is 32.

   Changing this setting does not affect users who are currently online. If the new limit is less than the number of online FTP users, no additional FTP users can log in until the number drops below the new limit.

   For more information about this command, see *Security Command Reference*.

Specifying an SSL server policy for SFTP connections

About this task

After you associate an SSL server policy with the device, a client that supports SFTP will establish a secure connection to the device to ensure data security.

Procedure

1. Enter system view.

   system-view

2. Associate an SSL server policy with the FTP server to ensure data security.

   ftp server ssl-server-policy *policy-name*

   By default, no SSL server policy is associated with the FTP server.

Setting the DSCP value for outgoing FTP packets

1. Enter system view.

```
system-view
```

2. Set the DSCP value for outgoing FTP packets.

   IPv4:

   ftp server dscp *dscp-value*

   IPv6:

   ftp server ipv6 dscp *dscp-value*

   By default, the DSCP value is 0.

Manually releasing FTP connections

To manually release FTP connections, execute the following commands in user view:

- Release the FTP connection established by using a specific user account:

  free ftp user *username*

- Release the FTP connection to a specific IP address:

  free ftp user-ip [**ipv6**] *ip-address* [**port** *port*]

Display and maintenance commands for the FTP server

Execute **display** commands in any view.

Task	Command
Display FTP server configuration and status information.	**display ftp-server**
Display detailed information about online FTP users.	**display ftp-user**

Using the device as an FTP client

FTP client configuration tasks at a glance

To use the device as an FTP server, perform the following tasks:

1. Establishing an FTP connection
2. (Optional.) Displaying command help information
3. (Optional.) Displaying directories and files on the FTP server
4. (Optional.) Managing directories on the FTP server
5. (Optional.) Working with files on the FTP server
6. (Optional.) Changing to another user account
7. (Optional.) Maintaining and troubleshooting the FTP connection
8. (Optional.) Terminating the FTP connection

Establishing an FTP connection

FTP connection establishment task list

To establish an FTP connection, perform the following tasks:

1. (Optional.) Specifying a source IP address for outgoing FTP packets

2. Establishing an FTP connection
3. Setting the FTP file transfer mode and operation mode

Restrictions and guidelines

The source IP address specified in the **ftp** command takes precedence over the one set by the **ftp client source** command.

The source IP address specified in the **ftp ipv6** command takes precedence over the one set by the **ftp client ipv6 source** command.

Specifying a source IP address for outgoing FTP packets

1. Enter system view.
   system-view
2. Specify a source IP address for outgoing FTP packets.
   IPv4:
   ftp client source { **interface** *interface-type interface-number* | **ip** *source-ip-address* }
   By default, no source IP address is specified. The device uses the primary IP address of the output interface as the source IP address.
   IPv6:
   ftp client ipv6 source { **interface** *interface-type interface-number* | **ipv6** *source-ipv6-address* }
   By default, no source IPv6 address is specified. The source address is automatically selected as defined in RFC 3484.

Establishing an FTP connection

- Log in to the FTP server from user view.
  IPv4:
  ftp [*ftp-server* [*service-port*] [**vpn-instance** *vpn-instance-name*] [**dscp** *dscp-value* | **source** { **interface** *interface-type interface-number* | **ip** *source-ip-address* }]] *
  IPv6:
  ftp ipv6 [*ftp-server* [*service-port*] [**vpn-instance** *vpn-instance-name*] [**dscp** *dscp-value* | **source** { **ipv6** *source-ipv6-address* | **interface** *interface-type interface-number* }] * [**-i** *interface-type interface-number*]]
- Log in to the FTP server from FTP client view.
  a. Enter FTP client view.
     ftp [**ipv6**]
  b. Log in to the FTP server.
     open *server-address* [*service-port*]

Setting the FTP file transfer mode and operation mode

1. Enter FTP client view from user view.
   ftp
2. Set the file transfer mode.
   - Set the file transfer mode to ASCII.
     ascii
   - Set the file transfer mode to binary.

binary

The default file transfer mode is binary.

3. Change the FTP operation mode.

   passive

   The default FTP operation mode is passive.

Displaying command help information

1. Enter FTP client view from user view.

   ftp

2. Display command help information.
   - **help** [*command-name*]
   - **?** [*command-name*]

Displaying directories and files on the FTP server

1. Enter FTP client view from user view.

   ftp

2. Display directories and files on the FTP server.
   - **dir** [*remotefile* [*localfile*]]
   - **ls** [*remotefile* [*localfile*]]

Managing directories on the FTP server

Prerequisites

Use the **dir** or **ls** command to display the directories and files on the FTP server.

Procedure

1. Enter FTP client view from user view.

   ftp

2. Manage directories on the FTP server.
   - Display the working directory that is being accessed on the FTP server.

     pwd
   - Change the working directory on the FTP server.

     cd { *directory* | **..** | **/** }
   - Return to the upper level directory on the FTP server.

     cdup
   - Create a directory on the FTP server.

     mkdir *directory*
   - Delete a directory from the remote FTP server.

     rmdir *directory*

CAUTION:

Delete a directory from the FTP server with caution. When you delete a directory from the FTP server, make sure the directory is no longer in use.

Managing directories on the FTP client

1. Enter FTP client view from user view.

   ftp

2. Display or change the local working directory of the FTP client.

   lcd [*directory* | **/**]

   To upload a file, use this command to change to the directory where the file resides. Downloaded files are saved in the working directory.

Working with files on the FTP server

Prerequisites

Use the **dir** or **ls** command to display the directories and files on the FTP server.

Procedure

1. Enter FTP client view from user view.

   ftp

2. Work with files on the FTP server.
   - Delete a file from the FTP server permanently.

     delete *remotefile*

     △ **CAUTION:**

     Permanently delete a file from the FTP server with caution. When you delete a file from the FTP server permanently, make sure the file is no longer in use.

     This command requires that you have the delete right.
   - Rename a file.

     rename [*oldfilename* [*newfilename*]]
   - Upload a file to the FTP server.

     put *localfile* [*remotefile*]
   - Download a file from the FTP server.

     get *remotefile* [*localfile*]
   - Add the content of a file on the FTP client to a file on the FTP server.

     append *localfile* [*remotefile*]
   - Specify the retransmit marker.

     restart *marker*

     Use this command together with the **put**, **get**, or **append** command.
   - Update a local file.

     newer *remotefile*
   - Get the missing part of a file.

     reget *remotefile* [*localfile*]

Changing to another user account

About this task

If you tried to access an FTP server but failed to pass the authentication, you can use this command to try again before the connection to the FTP server expires.

Restrictions and guidelines

For successful account change, you must enter the new username and password correctly. A wrong username or password can cause the FTP connection to be disconnected.

Procedure

1. Enter FTP client view from user view.

   `ftp`

2. Initiate an FTP authentication on the current FTP connection.

   `user` *`username`* `[` *`password`* `]`

Maintaining and troubleshooting the FTP connection

About this task

After you use the device to establish an FTP connection to the FTP server, use the commands in this section to locate and troubleshoot problems about the FTP connection.

Procedure

1. Enter FTP client view from user view.

   `ftp`

2. Maintain and troubleshoot the FTP connection.
   - Display FTP commands supported by the FTP server.

     `rhelp`
   - Display help information about an FTP command that is supported by the FTP server.

     `rhelp` *`protocol-command`*
   - Display FTP server status.

     `rstatus`
   - Display detailed information about a directory or file on the FTP server.

     `rstatus` *`remotefile`*
   - Display FTP connection status.

     `status`
   - Display the system information of the FTP server.

     `system`
   - Enable or disable FTP operation information display.

     `verbose`

     By default, this function is enabled.
   - Enable FTP client debugging.

     `debug`

     By default, FTP client debugging is disabled.
   - Clear the reply information in the buffer.

     `reset`

Terminating the FTP connection

1. Enter FTP client view from user view.

   ```
   ftp
   ```

2. Terminate the connection.
   - Terminate the connection to the FTP server without exiting FTP client view.

     ```
     disconnect
     close
     ```

   - Terminate the connection to the FTP server and return to user view.

     ```
     bye
     quit
     ```

Display and maintenance commands for the FTP client

Execute the **display** command in any view.

Task	Command
Display source IP address information on the FTP client.	**display ftp client source**

Configuring TFTP

About TFTP

Trivial File Transfer Protocol (TFTP) is a simplified version of FTP for file transfer over secure reliable networks. TFTP uses UDP port 69 for data transmission. In contrast to TCP-based FTP, TFTP does not require authentication or complex message exchanges, and is easier to deploy. TFTP is suited for reliable network environments.

The device can act as the TFTP server or TFTP client.

Restrictions and guidelines: TFTP configuration

You can upload a file from a TFTP client to the TFTP server or download a file from the TFTP server to a TFTP client. As a best practice, specify a file name that has not been used in the destination directory. If the destination file name has been used in the destination directory, the device deletes the existing file before saving the new file.

Configuring the device as a TFTP server

1. Enter system view.

`system-view`

2. Enable the TFTP server.

`tftp server enable`

By default, the TFTP server is disabled.

The device can act as the TFTP server on both IPv4 and IPv6 networks.

3. Set the TFTP server working directory.

`tftp server work-directory` *`directory`*

By default, the TFTP server working directory is the root directory of the default file system.

Configuring and using the IPv4 TFTP client

1. Enter system view.

`system-view`

2. (Optional.) Use an ACL to control the client's access to TFTP servers.

`tftp-server acl` *`acl-number`*

By default, no ACL is used for access control.

3. Specify the source IP address for TFTP packets sent by the TFTP client.

`tftp client source` { **`interface`** *`interface-type interface-number`* | **`ip`** *`source-ip-address`* }

By default, no source IP address is specified. The device uses the primary IP address of the output interface as the source IP address.

4. Return to user view.

`quit`

5. Download or upload a file in an IPv4 network.

tftp *tftp-server* { **get** | **put** | **sget** } *source-filename* [*destination-filename*] [**vpn-instance** *vpn-instance-name*] [**dscp** *dscp-value* | **source** { **interface** *interface-type interface-number* | **ip** *source-ip-address* }] *

The source IP address specified in this command takes precedence over the source IP address set by using the **tftp client source** command.

Configuring and using the IPv6 TFTP client

1. Enter system view.

   system-view

2. (Optional.) Use an ACL to control the client's access to TFTP servers.

   tftp-server ipv6 acl *ipv6-acl-number*

   By default, no ACL is used for access control.

3. Specify the source IPv6 address for TFTP packets sent by the TFTP client.

   tftp client ipv6 source { **interface** *interface-type interface-number* | **ipv6** *source-ipv6-address* }

   By default, no source IPv6 address is specified. The source address is automatically selected as defined in RFC 3484.

4. Return to user view.

   quit

5. Download or upload a file in an IPv6 network.

   tftp ipv6 *tftp-server* [**-i** *interface-type interface-number*] { **get** | **put** | **sget** } *source-filename* [*destination-filename*] [**vpn-instance** *vpn-instance-name*] [**dscp** *dscp-value* | **source** { **interface** *interface-type interface-number* | **ipv6** *source-ipv6-address* }] *

   The source IP address specified in this command takes precedence over the one set by the **tftp client ipv6 source** command.

Contents

Managing file systems

This chapter describes how to manage file systems.

About file system management

Storage media and file systems

The device supports both fixed (the flash memory) and hot swappable (USB disk and hard disk) storage media.

- The fixed storage medium has one file system.
- The hot swappable storage media can be partitioned. Each unpartitioned storage medium has one file system. On a partitioned storage medium, each partition has one file system.

Storage medium and file system naming conventions

The file system on the flash memory has the same name as the flash memory. The name has the following parts:

- Storage medium type **flash**.
- Colon (:).

A hard disk or USB disk name and the file system names share the following parts:

- Storage medium type, for example, **usb** for the USB disk.
- Sequence number, a lower-case English letter such as a, b, or c.
- Partition number, a digit that starts at 0 and increments by 1. If the storage medium is not partitioned, the system determines that the storage medium has one partition. (A storage medium name does not contain a partition number.)
- Colon (:).

IMPORTANT:

File system names are case sensitive and must be entered in lower case.

File system location

To identify a file system on the master device, you do not need to specify the file system location. To identify a file system on a subordinate member device, you must specify the file system location in the **slot***n***#** format. The *n* argument represents the IRF member ID of the member device. For example, the location is **slot2#** for a file system that resides on member device 2.

IMPORTANT:

The file system location string is case sensitive and must be entered in lower case.

Default file system

You are working with the default file system by default after you log in. To specify a file or directory on the default file system, you do not need to specify the file system name. For example, you do not need to specify any location information if you want to save the running configuration to the root directory of the default file system.

To change the default file system, use the BootWare or Boot ROM menu. For more information, see the software release notes.

Directories

Directories in a file system are structured in a tree form.

Root directory

The root directory is represented by a forwarding slash (/). For example, **flash:/** represents the root directory of the flash memory.

Working directory

The working directory is also called the current directory.

Directory naming conventions

When you specify a name for a directory, follow these conventions:

- A directory name can contain letters, digits, and special characters except for asterisks (*), vertical bars (|), forward slashes (/), backward slashes (\), question marks (?), left angle brackets (<), right angle brackets (>), quotation marks ("), and colons (:).
- A directory whose name starts with a dot character (.) is a hidden directory. To prevent the system from hiding a directory, make sure the directory name does not start with a dot character.

Commonly used directories

The device has some factory-default directories. The system automatically creates directories during operation. These directories include:

- **diagfile**—Stores diagnostic information files.
- **license**—Stores license files.
- **logfile**—Stores log files.
- **seclog**—Stores security log files.
- **versionInfo**—Stores software version information files.

Files

File naming conventions

When you specify a name for a file, follow these conventions:

- A file name can contain letters, digits, and special characters except for asterisks (*), vertical bars (|), forward slashes (/), backward slashes (\), question marks (?), left angle brackets (<), right angle brackets (>), quotation marks ("), and colons (:).
- A file whose name starts with a dot character (.) is a hidden file. To prevent the system from hiding a file, make sure the file name does not start with a dot character.

Common file types

The device has some factory-default files and might create some files automatically during operation. The types of these files include:

- **.ipe file**—Compressed software image package file.
- **.bin file**—Software image file.
- **.cfg file**—Configuration file.
- **.mdb file**—Binary configuration file.
- **.log file**—Log file.

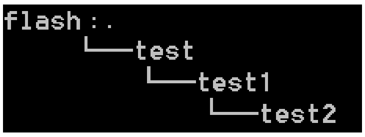
Specifying a directory name or file name

Specifying a directory name

To specify a directory, you can use the absolute path or a relative path. For example, the working directory is **flash:/**. To specify the **test2** directory in Figure 1, you can use the following methods:

- **flash:/test/test1/test2** (absolute path)
- **flash:/test/test1/test2/** (absolute path)
- **test/test1/test2** (relative path)
- **test/test1/test2/** (relative path)

Figure 1 Sample directory hierarchy



```
flash:.
    └──test
         └──test1
              └──test2
```

Specifying a file name

To specify a file, use the following methods:

- Enter the absolute path of the file and the file name in the format of *filesystem/directory1/directory2/…/directoryn/filename*, where *directoryn* is the directory in which the file resides.
- Enter the relative path of the file and the file name.

For example, the working directory is **flash:/**. The **samplefile.cfg** file is in the **test2** directory shown in Figure 1. To specify the file, you can use the following methods:

- **flash:/test/test1/test2/samplefile.cfg**
- **test/test1/test2/samplefile.cfg**

Restrictions and guidelines: File system management

To avoid file system corruption, do not perform the following tasks during file system management:

- Install or remove storage media.
- Perform a master/subordinate switchover.

If you remove a storage medium while a directory or file on the medium is being accessed, the device might not recognize the medium when you reinstall it. To reinstall this kind of storage medium, first complete one of the following tasks:

- If you were accessing a directory on the storage medium, change the working directory.
- If you were accessing a file on the storage medium, close the file.
- If another administrator was accessing the storage medium, unmount all partitions on the storage medium.

Make sure a USB disk is not write protected before an operation that requires the write right on the disk.

For the device to recognize a new hard disk or USB disk, you must first partition and format the hard disk or USB disk. To partition a disk, use the **`fdisk`** command. To format a disk, use the **`format`** command.

You cannot access a storage medium that is being partitioned, or a file system that is being formatted or repaired.

Before managing file systems, directories, and files, make sure you know the possible impact.

Managing storage media and file systems

Partitioning a storage medium

About this task

A storage medium can be divided into logical devices called partitions. Operations on one partition do not affect the other partitions.

Restrictions and guidelines

IMPORTANT:

Partitioning a storage medium clears all data on the medium.

The flash memory does not support partitioning.

Before partitioning a storage medium, make sure no other users are accessing the medium.

To partition a USB disk, make sure the disk is not write protected. If the disk is write protected, the partition operation will fail. To restore access to the USB disk, you must reinstall the disk or remount the file systems on the disk.

A partition must have a minimum of 32 MB of storage space.

The actual partition size and the specified partition size might have a difference of less than 5% of the storage medium's total size.

Prerequisites

Back up the files in the storage medium.

Procedure

To partition a storage medium, execute the following command in user view:

`fdisk` *`medium`* `[` *`partition-number`* `]`

This command is supported only on the default context.

To partition a storage medium evenly, specify the *`partition-number`* argument. To customize the sizes of partitions, do not specify the *`partition-number`* argument. The command will require you to specify a size for each partition.

Mounting or unmounting a file system

Restrictions and guidelines

You can mount or unmount only a file system that is on a hot-swappable storage medium.

You can unmount a file system only when no other users are accessing the file system.

To prevent a USB disk and the USB interface from being damaged, make sure the following requirements are met before unmounting file systems on the USB disk:

- The system has recognized the USB disk.
- The USB disk LED is not blinking.

Mounting a file system

To mount a file system, execute the following command in user view:

```
mount filesystem
```

This command is supported only on the default context.

File systems on a hot-swappable storage medium are automatically mounted when the storage medium is connected to the device. If the system cannot recognize a file system, you must mount the file system before you can access it.

Unmounting a file system

To unmount a file system, execute the following command in user view:

```
umount filesystem
```

This command is supported only on the default context.

To remove a hot-swappable storage medium from the device, you must first unmount all file systems on the storage medium to disconnect the medium from the device. Removing a connected hot-swappable storage medium might damage files on the storage medium or even the storage medium itself.

Formatting a file system

Restrictions and guidelines

You can format a file system only when no other users are accessing the file system.

Procedure

To format a file system, execute the following command in user view:

```
format filesystem [ ext4 | vfat ]
```

△ **CAUTION:**

Formatting a file system permanently deletes all files and directories in the file system. You cannot restore the deleted files or directories. If a startup configuration file exists in the file system, back up the file if necessary.

This command is supported only on the default context.

Repairing a file system

Restrictions and guidelines

If part of a file system is inaccessible, use this task to examine and repair the file system.

You can repair a file system only when no other users are accessing the file system.

Procedure

To repair a file system, execute the following command in user view:

```
fixdisk filesystem
```

This command is supported only on the default context.

Managing files and directories

Setting the operation mode for files and directories

About this task

The device supports the following operation modes:

- **alert**—The system prompts for confirmation when your operation might cause problems such as file corruption or data loss. This mode provides an opportunity for you to cancel a disruptive operation.
- **quiet**—The system does not prompt for confirmation when you perform a file or directory operation except the recycle bin emptying operation.

Procedure

1. Enter system view.

   `system-view`

2. Set the operation mode for files and directories.

   `file prompt` { **`alert`** | **`quiet`** }

   The default mode is alert.

Displaying file and directory information

To display file and directory information, execute the following command in user view:

`dir` [**`/all`**] [*`file`* | *`directory`* | **`/all-filesystems`**]

If multiple users perform file operations (for example, creating or deleting files or directories) at the same time, the output from this command might be incorrect.

Displaying the contents of a text file

To display the contents of a text file, execute the following command in user view:

`more` *`file`*

Displaying the working directory

To display the working directory, execute the following command in user view:

`pwd`

Changing the working directory

About this task

The default working directory is the root directory of the default file system on the master device.

Procedure

To change the working directory, execute the following command in user view:

`cd` { *`directory`* | **`..`** }

Creating a directory

To create a directory, execute the following command in user view:

mkdir *directory*

Renaming a file or directory

To rename a file or directory, execute the following command in user view:

rename { *source-file* | *source-directory* } { *dest-file* | *dest-directory* }

Copying a file

About this task

You can copy a file as follows:

- Copy a local file and save it locally.
- Copy a local file and save it to a remote file server.
- Copy a file from a remote file server and save it locally.

The remote file server can be an FTP, TFTP, or HTTP server.

Restrictions and guidelines

To specify a file or directory on a remote file server, use the following guidelines:

Location	Name format	Remarks
On an FTP server	Enter the URL in the format of **ftp://***FTP username*[**:***password*]**@***server address*[**:***port number*]**/***file path*[**/***file name*].	The username and password must be the same as the username and password configured on the FTP server. If the server authenticates users only by the username, you are not required to enter the password. For example, to use the username **a** and password **1** and specify the **startup.cfg** file in the authorized working directory on the FTP server 1.1.1.1, enter **ftp://a:1@1.1.1.1/startup.cfg**.
On a TFTP server	Enter the URL in the format of **tftp://***server address*[**:***port number*]**/***file path*[**/***file name*].	For example, to specify the **startup.cfg** file in the working directory on TFTP server 1.1.1.1, enter the URL **tftp://1.1.1.1/startup.cfg**.
On an HTTP server	Enter the URL in the format of **http://**[*HTTP username*[**:***password*]@]*server address*[**:***port number*]**/***filepath*[**/***file name*].	The username and password in the URL must be the same as the username and password configured on the server. If only the username is required for authentication, you do not need to enter the password. If authentication is not required, you do not need to enter the username or password. For example, the **startup.cfg** file is saved in the authorized directory on the HTTP server at 1.1.1.1. The HTTP account username and password are **a** and **1**, respectively. To copy the file, enter the URL **http://a:1@1.1.1.1/startup.cfg**. If authentication is not required, enter the URL **http://1.1.1.1/startup.cfg**.

To specify an IPv6 address, enclose the IPv6 address in square brackets ([ ]), for example, **ftp://test:test@[2001::1]:21/test.cfg**.

Procedure

To copy a file, execute the command in user view.

copy *source-file* { *dest-file* | *dest-directory* } [**vpn-instance** *vpn-instance-name*] [**source interface** *interface-type interface-number*] [**append**]

Moving a file

To move a file, execute the following command in user view:

move *source-file* { *dest-file* | *dest-directory* }

Deleting and restoring files

About this task

You can delete a file permanently or move it to the recycle bin of the file system. A file moved to the recycle bin can be restored, but a permanently deleted file cannot.

Each file system has a recycle bin. A recycle bin is a directory named **.trash** in the root directory of the file system.

Restrictions and guidelines

Files in the recycle bin occupy storage space. To release the occupied storage space, delete files from the recycle bin.

To delete files from the recycle bin, use the **reset recycle-bin** command. If you use the **delete** command, the recycle bin might not be able to operate correctly.

To display files in a recycle bin, use one of the following methods:

- Access the root directory of the file system and execute the **dir /all .trash** command.
- Access the recycle bin directory of the file system and execute the **dir** command.

Deleting a file

To delete a file, execute one of the following commands in user view:

- Delete a file by moving it to the recycle bin.
  delete *file*
- Delete a file permanently.
  delete /unreserved *file*

△ **CAUTION:**

The **delete /unreserved** *file* command deletes a file permanently. The file cannot be restored.

- Delete files from the recycle bin.
  reset recycle-bin [**/force**]

△ **CAUTION:**

The files in a recycle bin can be restored by using the **undelete** command. If you delete a file from the recycle bin, that file cannot be restored. Before you delete files from a recycle bin, make sure the files are no longer in use.

Restoring a file

To restore a file from the recycle bin, execute the following command in user view:

undelete *file*

Deleting a directory

To delete a directory, execute the following command in user view:

rmdir *directory*

△ **CAUTION:**

To delete a directory, you must first delete all files and subdirectories in the directory permanently or move them to the recycle bin. If you move them to the recycle bin, executing the **rmdir** command to delete the directory will delete them permanently. Before you use the **rmdir** command to delete a directory, you must make sure the directory and its files and subdirectories are no longer in use.

Archiving files and directories

About this task

You can archive files and directories to a single file for purposes such as file backup. The original files and directories still exist.

When you archive files and directories, you can choose to compress the archive files so the archive files use less storage space.

Procedure

To archive files and directories, execute the following command in user view:

tar create [**gz**] **archive-file** *dest-file* [**verbose**] **source** { *source-file* | *source-directory* }&<1-5>

Extracting files and directories

About this task

Use this feature to extract files and directories from archive files.

Restrictions and guidelines

To specify the **screen** keyword for the **tar extract** command, first use the **tar list** command to identify the types of the archived files. As a best practice, specify the keyword only if all archived files are text files. Displaying the content of an archived non-text file that contains terminal control characters might result in garbled characters and even cause the terminal unable to operate correctly. To use the terminal again, you must close the current connection and log in to the device again.

Procedure

To extract files and directories, execute the following commands in user view:

1. (Optional.) Display archived files and directories.

   tar list archive-file *file*

2. Extract files and directories.

   tar extract archive-file *file* [**verbose**] [**screen** | **to** *directory*]

Compressing a file

To compress a file, execute the following command in user view:

gzip *file*

Decompressing a file

To decompress a file, execute the following command in user view:

gunzip *file*

Calculating the file digest

About this task

File digests are used to verify file integrity.

Procedure

To calculate the digest of a file, execute one of the following commands in user view:

- Use the SHA-256 algorithm.

  sha256sum *file*
- Use the MD5 algorithm.

  md5sum *file*

Displaying processes that are using a file system, directory, or file

About this task

Execute this command if you fail to execute a command such as **fdisk**, **fixdisk**, **format**, **umount**, **rmdir**, **rename**, **delete**, or **copy** command for a file system, directory, or file. View the command output to identify whether a process is using the file system, directory, or file.

Procedure

To display processes that are using a file system, directory, or file, execute the following command in user view:

fuser { *directory* | *file* | *filesystem* }

Contents

Managing configuration files

About configuration file management

You can manage configuration files from the CLI or the BootWare menu. The following information explains how to manage configuration files from the CLI.

A configuration file saves a set of commands for configuring software features on the device. You can save any configuration to a configuration file so the configuration can survive a reboot. You can also back up configuration files to a host for future use.

Configuration types

Initial configuration

Initial configuration is the collection of initial default settings for the configuration commands in software.

The device starts up with the initial configuration if you access the BootWare menu and select the **Skip Current System Configuration** option. In this situation, the device might also be described as starting up with empty configuration.

No commands are available to display the initial configuration. To view the initial default settings for the configuration commands, see the Default sections in the command references.

Factory defaults

Factory defaults are custom basic settings that came with the device. Factory defaults vary by device models and might differ from the initial default settings for the commands.

The device starts up with the factory defaults if no next-startup configuration files are available.

To display the factory defaults, use the **`display default-configuration`** command.

Startup configuration

The device uses startup configuration to configure software features during startup. After the device starts up, you can specify the configuration file to be loaded at the next startup. This configuration file is called the next-startup configuration file. The configuration file that has been loaded is called the current startup configuration file.

You can display the startup configuration by using one of the following methods:

- To display the contents of the current startup configuration file, execute the **`display current-configuration`** command before changing the configuration after the device reboots.
- To display the contents of the next-startup configuration file, use the **`display saved-configuration`** command.
- Use the **`display startup`** command to display names of the current startup configuration file and next-startup configuration files. Then, you can use the **`more`** command to display the contents of the specified startup configuration file.

Running configuration

The running configuration includes unchanged startup settings and new settings. The running configuration is stored in memory and is cleared at a device reboot or power off. To use the running configuration after a power cycling or reboot, save it to a configuration file.

To display the running configuration, use the **`display current-configuration`** command.

Configuration file types and file selection process at startup

When you save the configuration, the system saves the settings to a .cfg configuration file and to an .mdb file.

- A .cfg configuration file is a human-readable text file and its contents can be displayed by using the **more** command. Configuration files you specify for saving the configuration must use the .cfg extension.
- An .mdb file is a user-inaccessible binary file that has the same name as the .cfg file. The device loads an .mdb file faster than loading a .cfg file.

At startup, the device uses the following procedure to identify the configuration file to load:

1. The device searches for a valid .cfg next-startup configuration file. For more information about the file selection rules, see "Next-startup configuration file redundancy."
2. If a valid .cfg next-startup configuration file is found, the device searches for an .mdb file that has the same name and checksum as the .cfg file.
3. If a matching .mdb file is found, the device starts up with the .mdb file. If none is found, the device starts up with the .cfg file.

If no .cfg next-startup configuration files are available, the device starts up with the factory defaults.

Unless otherwise stated, the term "configuration file" in this document refers to a .cfg configuration file.

Next-startup configuration file redundancy

You can specify one main next-startup configuration file and one backup next-startup configuration file for redundancy.

At startup, the device tries to select the .cfg startup configuration in the following order:

1. The main next-startup configuration file.
2. The backup next-startup configuration file if the main next-startup configuration file does not exist or is corrupt.

If no next-startup configuration files are available, the device starts up with the factory defaults.

Configuration file content organization and format

IMPORTANT:

To run on the device, a configuration file must meet the content and format requirements. To ensure a successful configuration load at startup, use a configuration file created on the device. If you edit the configuration file, make sure all edits are compliant with the requirements.

A configuration file must meet the following requirements:

- All commands are saved in their complete form.
- Commands are sorted into sections by different command views, including system view, interface views, protocol views, and user line views.
- Two adjacent sections are separated by a pound sign (#).
- The configuration file ends with the word **return**.

The following is a sample configuration file excerpt:

```
#
local-user root class manage
```

```
 password hash
$h$6$Twd73mLrN8O2vvD5$Cz1vgdpR4KoTiRQNE9pg33gU14Br2p1VguczLSVyJLO2huV5Syx/LfDIf8ROLtV
ErJ/C31oq2rFtmNuyZf4STw==
 service-type ssh telnet terminal
 authorization-attribute user-role network-admin
 authorization-attribute user-role network-operator
#
interface Vlan-interface1
 ip address 192.168.1.84 255.255.255.0
#
```

Configuration rollback

Configuration rollback allows you to replace the running configuration with the configuration in a configuration file without rebooting the device. You can use this feature for the following purposes:

- Reverting to a previous configuration state.
- Adapting the running configuration to different network environments.

Restrictions and guidelines: Configuration file management

For the device to load the configuration correctly at startup, do not save the factory defaults or startup configuration files on a removable hard disk.

Enabling configuration encryption

About this task

Configuration encryption enables the device to encrypt a startup configuration file automatically when it saves the running configuration. All devices running NF software use the same method to encrypt configuration files.

Restrictions and guidelines

Any devices running NF software can decrypt the encrypted configuration files. To prevent an encrypted file from being decoded by unauthorized users, make sure the file is accessible only to authorized users.

You cannot use the **more** command to view the contents of an encrypted configuration file.

You cannot compare an encrypted configuration file with any configuration for their differences.

Procedure

1. Enter system view.

   system-view

2. Enable configuration encryption.

   configuration encrypt { **private-key** | **public-key** }

   By default, configuration encryption is disabled.

Saving the running configuration

About this task

When you save the running configuration to a .cfg configuration file, you can specify the file as a next-startup configuration file or not.

If you are specifying the file as a .cfg next-startup configuration file, use one of the following methods to save the configuration:

- **Fast mode**—Use the **save** command without the **safely** keyword. In this mode, the device directly overwrites the target next-startup configuration file. If a reboot or power failure occurs during this process, the next-startup configuration file is lost. You must specify a new startup configuration file after the device reboots (see "Specifying a next-startup configuration file").
- **Safe mode**—Use the **save** command with the **safely** keyword. Safe mode is slower than fast mode, but more secure. In safe mode, the system saves the configuration in a temporary file and starts overwriting the target next-startup configuration file after the save operation is complete. If a reboot or power failure occurs during the save operation, the next-startup configuration file is still retained. Use the safe mode if the power source is not reliable or you are remotely configuring the device.

Restrictions and guidelines

To prevent the loss of next-startup configuration, make sure no one reboots or power cycles the device while the device is saving the running configuration.

When an IRF member device splits from the IRF fabric, its settings are retained in memory but removed from the running configuration on the IRF fabric. Saving the running configuration before the IRF fabric recovers will remove the member device's settings from the next-startup configuration file.

If you have saved the running configuration before the member device rejoins the IRF fabric, perform the following steps to restore the member device settings to the next-startup configuration file:

1. Resolve the split issue.
2. Reboot the member device to rejoin the IRF fabric.
3. After the member device rejoins the IRF fabric, execute the **display current-configuration** command to verify that the member device's settings have been restored from memory to the running configuration.
4. Save the running configuration to the next-startup configuration file on the IRF fabric.

IMPORTANT:

To ensure a successful configuration restoration, make sure the IRF fabric has not rebooted after the member device left.

Procedure

To save the running configuration, perform one of the following tasks in any view:

- Save the running configuration to a configuration file without specifying the file as a next-startup configuration file.

  save *file-url* [**all** | **slot** *slot-number*]
- Save the running configuration to a configuration file in the root directory of the storage medium on each IRF member device, and specify the file as a next-startup configuration file.

  save [**safely**] [**backup** | **main**] [**force**] [**context-all** | **changed**]

  As a best practice, specify the **safely** keyword for reliable configuration saving.

CAUTION:

Use caution when you save the running configuration. This operation will overwrite the settings in the target configuration file. When you perform this operation, carefully read the messages displayed by the system and make sure you fully understand the impact of the operation on services.

Comparing configurations for their differences

About this task

You can compare configuration files or compare a configuration file with the running configuration for their differences.

If you specify the next-startup configuration for a comparison, the system selects the next-startup configuration file to be compared with in the following order:

1. The main next-startup configuration file.
2. The backup next-startup configuration file if the main next-startup configuration file is unavailable.

If both configuration files are unavailable, the system displays a message indicating that no next-startup configuration files exist.

Restrictions and guidelines

If you specify a configuration file for a comparison, the configuration file must be a .cfg configuration file.

Procedure

To compare configurations for their differences, perform one of the following tasks in any view:

- Display the differences that a configuration file, the running configuration, or the next-startup configuration has as compared with the specified source configuration file.

  display diff configfile *file-name-s* { **configfile** *file-name-d* | **current-configuration** | **startup-configuration** }
- Display the differences that a configuration file or next-startup configuration has as compared with the running configuration.

  display diff current-configuration { **configfile** *file-name-d* | **startup-configuration** }
- Display the differences that a configuration file has as compared with the next-startup configuration.

  display diff startup-configuration configfile *file-name-d*
- Display the differences that the running configuration has as compared with the next-startup configuration.
  - Method 1:

    display diff startup-configuration current-configuration
  - Method 2:

    display current-configuration diff

Archiving the running configuration

About running configuration archiving

You can perform local or remote configuration archiving to archive the running configuration to a directory on the device or to a remote server, respectively.

The remote server can be an FTP, TFTP, or SCP server.

The following are methods for archiving the running configuration:

- **Automatic configuration archiving**—The system automatically archives the running configuration at intervals as configured.
- **Manual configuration archiving**—You can disable automatic configuration archiving and manually archive the running configuration if the configuration will be rarely changed. You can also use this method to back up configuration before performing complicated configuration tasks.

Restrictions and guidelines for archiving the running configuration

Local archiving (the **`archive configuration location`** command) and remote archiving (the **`archive configuration server`** command) are mutually exclusive. You cannot use the two features at the same time.

If you configure parameters multiple times for an archiving method, the most recent configuration takes effect.

Configuring local configuration archiving

About this task

Local configuration archives are named in the format of *prefix_serial number*.**cfg** (for example, **archive_1.cfg** and **archive_2.cfg**). The serial number is automatically assigned from 1 to 1000, increasing by 1. After the serial number reaches 1000, it restarts from 1.

If you change the file directory or file name prefix, the following events occur:

- The old configuration archives change to common configuration files.
- The configuration archive counter is reset. The serial number for new configuration archives starts at 1.
- The **`display archive configuration`** command no longer displays the old configuration archives.

The configuration archive counter does not restart when you delete configuration archives from the archive directory. However, if the device reboots after all configuration archives have been deleted, the configuration archive counter restarts. The serial number for new configuration archives starts at 1.

If an archiving has started based on the existing archive parameters when an archive parameter is changed, the archive will still be retained in the old directory. However, the **`display archive configuration`** command will not display the record about this archiving.

Restrictions and guidelines

After the maximum number of configuration archives is reached, the system deletes the oldest archive to make room for the new archive.

The **`undo archive configuration location`** command removes the local configuration archive directory and file name prefix settings, but it does not delete the configuration archives on the device. In addition, the command performs the following operations:

- Disables both the manual and automatic configuration archiving features.
- Restores the default settings for the **`archive configuration interval`** and **`archive configuration max`** commands.

- Clears the configuration archive information displayed by using the **display archive configuration** command.

The local archiving feature saves the running configuration only on the master device. To make sure the system can archive the running configuration after a master/subordinate switchover, create the configuration archive directory on all IRF members.

Procedure

1. Enter system view.

   system-view

2. Set the directory and file name prefix for archiving the running configuration to the local device.

   archive configuration location *directory* **filename-prefix** *filename-prefix*

   By default, no path or file name prefix is set for configuration archives on the device, and the system does not regularly archive the running configuration.

   In an IRF fabric, the configuration archive directory must already exist on the master device and cannot include a member ID.

3. (Optional.) Set the maximum number of configuration archives that can be stored on the device.

   archive configuration max *file-number*

   The default number is 5.

   Change the setting depending on the amount of storage available on the device.

4. Archive the running configuration to the configuration archive directory. Choose one of the following methods:

   - Enable automatic configuration archiving and set the archiving interval.

     archive configuration interval *interval*

     By default, automatic configuration archiving is disabled.

   - Execute the following commands in sequence to manually archive the running configuration in user view:

     quit

     archive configuration

Configuring remote configuration archiving

About this task

If you archive the running configuration to a remote server, configuration archives are named in the format of *prefix_YYYYMMDD_HHMMSS***.cfg** (for example, **archive_20170526_203430.cfg**).

Restrictions and guidelines

The **undo archive configuration server** command removes the remote configuration archive directory and file name prefix settings from the device, but it does not delete the configuration archives on the server. In addition, the command performs the following operations:

- Disables both the manual and automatic configuration archiving features.
- Restores the default setting for the **archive configuration server user** and **archive configuration server password** commands.
- Clears the configuration archive information displayed by using the **display archive configuration** command.

When you modify parameters (for example, the username or password) for remote archiving, make sure the changes are consistent between the device and the server. A manual or automatic remote archiving will fail if it has started before you change the device and the server settings to be consistent.

Procedure

1. Enter system view.

   system-view

2. Configure parameters for archiving the running configuration to a remote server.

   archive configuration server { **ftp** | **tftp** | **scp** } { *ipv4-address* | **ipv6** *ipv6-address* } [**port** *port-number*] [**vpn-instance** *vpn-instance-name*] [**directory** *directory*] **filename-prefix** *filename-prefix* [**interval** *interval*]

   By default, no parameters are configured for archiving the running configuration to a remote server.

   To automatically archive the running configuration regularly to the server, specify the **interval** *interval* option.

3. If an FTP or SCP server is used, configure the username and password for accessing the server.

   a. Configure the username.

      archive configuration server user *user-name*

      By default, no username is configured for accessing the FTP or SCP server.

   b. Configure the password.

      archive configuration server password { **cipher** | **simple** } *string*

      By default, no password is configured for accessing the FTP or SCP server.

   Make sure the username and password are the same as the FTP or SCP account settings on the FTP or SCP server.

4. To manually archive the running configuration to the remote server:

   a. Return to user view.

      quit

   b. Manually archive the running configuration to the remote server.

      archive configuration

Rolling back the running configuration

About configuration rollback

Configuration rollback allows you to replace the running configuration with the configuration in a replacement configuration file without rebooting the device. The replacement configuration file can be saved on the local device or on an FTP or TFTP server.

The configuration rollback feature compares the running configuration with the specified replacement configuration file and handles configuration differences as follows:

- If a command in the running configuration is not in the replacement file, the rollback feature executes the **undo** form of the command.
- If a command in the replacement file is not in the running configuration, the rollback feature adds the command to the running configuration.
- If a command has different settings in the running configuration and the replacement file, the rollback feature replaces the running command setting with the setting in the replacement file.

Restrictions and guidelines for configuration rollback

△ **CAUTION:**

A configuration rollback might cause service disruption.

To ensure a successful rollback, do not install or remove expansion interface cards or perform a master/subordinate switchover while the system is rolling back the configuration.

Make sure the replacement configuration file is created by using the configuration archive feature or the **save** command on the device. If the configuration file is not created on the device, make sure the command lines in the configuration file are fully compatible with the device.

If configuration rollback fails for some command lines, the system outputs a rollback failure message. To identify those command lines, use the **display diff current-configuration configfile** *file-name-d* command, with the replacement file specified for the *file-name-d* argument. The command lines that have failed to roll back are displayed as configuration differences between the running configuration and the replacement configuration file.

The configuration rollback feature might fail to reconfigure some commands in the running configuration for one of the following reasons:

- A command cannot be undone because prefixing the **undo** keyword to the command does not result in a valid **undo** command. For example, if the **undo** form designed for the **A** [**B**] **C** command is **undo A C**, the configuration rollback feature cannot undo the **A B C** command. This is because the system does not recognize the **undo A B C** command.
- A command (for example, a hardware-dependent command) cannot be deleted, overwritten, or undone due to system restrictions.
- The commands in different views are dependent on each other.
- Commands or command settings that the device does not support cannot be added to the running configuration.

In an HA group environment, you must execute the **configuration manual-sync** command to back up the configuration on the primary device to the secondary device for configuration consistency. For more information about HA groups, see *High Availability Configuration Guide*.

Performing a local configuration rollback

Restrictions and guidelines

The replacement configuration file must be stored unencrypted on the local system.

Procedure

1. Enter system view.

   system-view

2. Roll the running configuration back to the configuration in a configuration file.

   configuration replace file *filename*

   △ **CAUTION:**

   This operation will cause settings not in the replacement configuration file to be lost, which might cause service interruption. When you perform configuration rollback, make sure you fully understand its impact on your network.

Scheduling or performing a remote configuration rollback

About this task

You can roll back the running configuration by using a configuration file stored on an FTP or TFTP server.

When you perform this task, you can roll back the running configuration immediately or schedule a rollback for a future date and time.

The device performs a remote rollback as follows:

1. Downloads a replacement configuration file from a remote server.
2. Saves the downloaded file as a temporary file.
3. Replaces the running configuration with the configuration in the temporary file.
4. Deletes the temporary file after the configuration rollback finishes.

Restrictions and guidelines

The **undo configuration replace server** command performs the following operations:

- Disables the running configuration remote rollback feature.
- Restores the default settings for the **configuration replace server user** and **configuration replace server password** commands.
- Clears the configuration rollback information displayed by using the **display configuration replace server** command.

You can cancel a configuration rollback schedule anytime before its scheduled date and time. When you schedule a rollback, you can specify the date and time or specify only the time to execute the rollback.

- If you specify a rollback date with the rollback time, the specified date must be the same or later than the current system date. If the specified date is the same as the current system date, the specified time must be later than the current system time. After you create the rollback schedule, be careful with changing the system clock backward. The rollback schedule will be canceled automatically if it expires before it could be executed because the system date or time is changed backward.
- If you do not specify a rollback date with the rollback time, the device compares the specified rollback time with the current system time.
  - If the specified rollback time is later than the current system time, the device performs a rollback at the specified time on the current day.
  - If the specified rollback time is earlier than the current system time, the device performs a rollback at the specified time on the next day.

Procedure

1. Enter system view.

   system-view

2. Configure server parameters for remote configuration rollback.

   configuration replace server { **ftp** | **tftp** } { *ipv4-address* | **ipv6** *ipv6-address* } [**port** *port-number*] [**vpn-instance** *vpn-instance-name*] [**directory** *directory*] **file** *filename*

   By default, no server parameters are configured for remote configuration rollback.

   The configuration file specified by using this command will be used for rollback if no configuration file is specified when you perform or schedule a remote rollback.

3. If an FTP server is used, configure the username and password for accessing the remote FTP server.

a. Configure the username.

   configuration replace server user *user-name*

   By default, no username is configured for accessing the FTP server.

   If no username is configured, the username will be **anonymous**.

b. Configure the password.

   configuration replace server password { **cipher** | **simple** } *string*

   By default, no password is configured for accessing the FTP server.

Make sure the username and password are the same as the FTP account settings on the FTP server.

4. Roll the running configuration back to the configuration in a replacement configuration file.
   - Roll the running configuration back immediately.

     configuration replace server file [*filename*]
   - Schedule a configuration rollback.

     configuration replace server file [*filename*] **at** *time* [*date*]

   If no replacement configuration file is specified, the configuration file specified by using the **configuration replace server** command is used for rollback.

△ **CAUTION:**

This operation will cause settings not in the replacement configuration file to be lost, which might cause service interruption. When you perform configuration rollback, make sure you fully understand its impact on your network.

Specifying a next-startup configuration file

Restrictions and guidelines

△ **CAUTION:**

Using the **undo startup saved-configuration** command can cause an IRF split after the IRF fabric or an IRF member reboots. When you execute this command, make sure you understand its impact on your network.

As a best practice, specify different files as the main and backup next-startup configuration files.

The **undo startup saved-configuration** command changes the attribute of the main or backup next-startup configuration file to NULL instead of deleting the file.

Prerequisites

In an IRF fabric, make sure the specified configuration file is valid and has been saved to the root directory of a storage medium on each member device. In addition, make sure the storage media are the same type across all IRF member devices.

Procedure

1. Specify a next-startup configuration file. Choose one of the following methods:
   - Execute the following command in user view to specify a next-startup configuration file:

     startup saved-configuration *cfgfile* [**backup** | **main**]

     By default, no next-startup configuration files are specified.
   - Execute the following command in any view to save the running configuration to a file and specify the file as a next-startup configuration file:

     save [**safely**] [**backup** | **main**] [**force**]

For more information about this command, see "Saving the running configuration."

If you do not specify the **backup** or **main** keyword, this command specifies the configuration file as the main next-startup configuration file.

2. (Optional.) Verify the configuration. Use one of the following commands in any view:
   - Display the names of the configuration files for this startup and the next startup.
     display startup
   - Display the contents of the configuration file for the next system startup.
     display saved-configuration

Backing up and restoring the main next-startup configuration file

About backing up and restoring the main next-startup configuration file

You can back up the main next-startup configuration file to a TFTP server or restore the main next-startup configuration file from a TFTP server.

Prerequisites for configuration backup and restoration

Before you back up or restore the main next-startup configuration file, perform the following tasks:

- Make sure the following requirements are met:
  - The server is reachable.
  - The server is enabled with TFTP service.
  - You have read and write permissions to the server.
- For configuration backup, use the **display startup** command to verify that the main next-startup configuration file has been specified in user view. If no next-startup configuration file has been specified or the specified configuration file does not exist, the backup operation will fail.

Backing up the main next-startup configuration file to a TFTP server

To back up the main next-startup configuration file to a TFTP server, execute the following command in user view:

backup startup-configuration to { *ipv4-server* | **ipv6** *ipv6-server* } [*dest-filename*] [**vpn-instance** *vpn-instance-name*]

Restoring the main next-startup configuration file from a TFTP server

1. Restore the main next-startup configuration file from a TFTP server in user view.
   restore startup-configuration from { *ipv4-server* | **ipv6** *ipv6-server* } *src-filename* [**vpn-instance** *vpn-instance-name*]

2. (Optional.) Verify that the specified configuration file has been set as the main next-startup configuration file. Use one of the following commands in any view:
   - Display the names of the configuration files for this startup and the next startup.

     `display startup`
   - Display the contents of the configuration file for the next system startup.

     `display saved-configuration`

Deleting a next-startup configuration file

About this task

You can perform this task to delete a next-startup configuration file.

If both the main and backup next-startup configuration files are deleted, the device uses the factory defaults at the next startup.

To delete a file that is set as both main and backup next-startup configuration files, you must execute both the **`reset saved-configuration backup`** command and the **`reset saved-configuration main`** command. Using only one of the commands removes the specified file attribute instead of deleting the file.

For example, if the **`reset saved-configuration backup`** command is executed, the backup next-startup configuration file setting is set to NULL. However, the file is still used as the main file. To delete the file, you must also execute the **`reset saved-configuration main`** command.

Restrictions and guidelines

△ **CAUTION:**

This task permanently deletes a next-startup configuration file from all IRF member devices. As a best practice, make sure you have a configuration backup before you perform this task.

If you do not specify the **`backup`** or **`main`** keyword when you perform this task, the main next-startup configuration is deleted.

Procedure

To delete a next-startup configuration file, execute the following command in user view:

`reset saved-configuration` [**`backup`** | **`main`**]

Display and maintenance commands for configuration files

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display configuration archive information.	**`display archive configuration`**
Display information about remote configuration rollback.	**`display configuration replace server`**
Display the running configuration.	**`display current-configuration`** [**`configuration`** [*`module-name`*] | **`interface`** [*`interface-type`* [*`interface-number`*]] | **`vpn-instance`** [*`vpn-instance-name`*]]

Task	Command
Display the differences that the running configuration has as compared with the next-startup configuration.	**display current-configuration diff**
Display the factory defaults.	**display default-configuration**
Display the differences between configurations.	• **display diff configfile** *file-name-s* { **configfile** *file-name-d* | **current-configuration** | **startup-configuration** } • **display diff current-configuration** { **configfile** *file-name-d* | **startup-configuration** } • **display diff startup-configuration** { **configfile** *file-name-d* | **current-configuration** }
Display the contents of the configuration file for the next system startup.	**display saved-configuration**
Display the names of the configuration files for this startup and the next startup.	**display startup**
Display the valid configuration in the current view.	**display this**
Delete next-startup configuration files.	**reset saved-configuration** [**backup** | **main**]

Contents

Upgrading software

About software upgrade

Software upgrade enables you to upgrade a software version, add new features, and fix software bugs. This chapter describes software types and release forms, compares software upgrade methods, and provides the procedures for upgrading software from the CLI.

Software types

The following software types are available:

- **BootWare image**—Also called the Boot ROM image. This image contains a basic segment and an extended segment.
    - The basic segment is the minimum code that bootstraps the system.
    - The extended segment enables hardware initialization and provides system management menus. When the device cannot start up correctly, you can use the menus to load software and the startup configuration file or manage files.

  Typically, the BootWare image is integrated into the Boot image to avoid software compatibility errors.
- NF **image**—Includes the following image subcategories:
    - **Boot image**—A .bin file that contains the Linux operating system kernel. It provides process management, memory management, and file system management.
    - **System image**—A .bin file that contains the NF kernel and standard features, including device management, interface management, configuration management, and routing.
    - **Feature image**—A .bin file that contains advanced or customized software features. You can purchase feature images as needed.
    - **Patch image**—A .bin file that is released for fixing bugs without rebooting the device. A patch image does not add or remove features.

      Patch images have the following types:
        - **Incremental patch images**—A new incremental patch image can cover all, part, or none of the functions provided by an old incremental patch image. A new incremental patch image can coexist with an old incremental patch image on the device only when the former covers none of the functions provided by the latter.
        - **Non-incremental patch images**—A new non-incremental patch image covers all functions provided by an old non-incremental patch image. Each of the boot, system, and feature images can have one non-incremental patch image, and these patch images can coexist on the device. The device uninstalls the old non-incremental patch image before installing a new non-incremental patch image.

      An incremental patch image and a non-incremental patch image can coexist on the device.

  NF images that have been loaded are called current software images. NF images specified to load at the next startup are called startup software images.

BootWare image, boot image, and system image are required for the device to operate.

You can install up to 32 .bin files on the device, including one boot image file, one system image file, and up to 30 feature and patch image files.

Software release forms

Software images are released in one of the following forms:

- Separate .bin files. You must verify compatibility between software images.
- As a whole in one .ipe package file. The images in an .ipe package file are compatible. The system decompresses the file automatically, loads the .bin images and sets them as startup software images.

NOTE:

Software image file names use the *model-comware version-im*age *type-release* format. This document uses **boot.bin** and **system.bin** as boot and system image file names.

Upgrade methods

Upgrade method	Software types	Remarks
Upgrading from the CLI by using the boot loader method	• BootWare image • NF images (excluding patches)	This method is disruptive. You must reboot the entire device to complete the upgrade.
Performing an ISSU from the CLI	NF images	This method enables a software upgrade with a minimum amount of downtime. Use this method if possible. For more information about ISSU, see "Performing an ISSU."
Upgrading from the BootWare menu	• BootWare image • NF images	Use this method when the device cannot start up correctly. To use this method, first connect to the console port and power cycle the device. Then, press **Ctrl+B** at prompt to access the BootWare menu. For more information about upgrading software from the BootWare menu, see the release notes for the software version. **IMPORTANT:** Use this method only when you do not have any other choice.

This chapter covers only upgrading software from the CLI by using the boot loader method.

Software image loading

Startup software images

To upgrade software, you must specify the upgrade files as the startup software images for the device to load at next startup. You can specify two lists of software images: one main and one backup. The device first loads the main startup software images. If the main startup software images are not available, the devices loads the backup startup software images.

Image loading process at startup

At startup, the device performs the following operations after loading and initializing BootWare:

1. Loads main images.

2. If any main image does not exist or is invalid, loads the backup images.
3. If any backup image does not exist or is invalid, checks the main or backup boot image.
4. If both the main and backup boot images do not exist or are invalid, the device cannot start up.

Digitally signed software images

The software images for the device are digitally signed for authenticity and integrity verification. This mechanism ensures that the software installed on the system is from a trusted source and has not been tampered with in the transfer, storage, or installation phase.

The system performs software digital signature verification for authenticity and integrity in the following situations:

- Before the system loads a software image during startup. If the digital signature verification fails, the system will not load the image and you will receive a digital signature verification failure message.
- When you specify a software image to upgrade the device from the BootWare menu. If the digital signature verification fails, the system will not set the image for upgrade and you will receive a digital signature verification failure message.
- Before the system loads a BootWare image to the Normal area of BootWare. If the digital signature verification fails, the system will not load the image and you will receive a digital signature verification failure message.
- When you specify a software image as a startup image through the boot loader. The system will verify the digital signature of the image before it updates the startup image list with the specified image. If the digital signature verification fails, the system will not update the startup image list and you will receive a digital signature verification failure message.
- When you specify startup images for an ISSU. The system verifies the digital signature of a software image for authenticity and integrity before it sets and loads that image as a main startup image. If the digital signature verification fails, the system will not set or load the image as a main startup image and you will receive a digital signature verification failure message. For more information about ISSU, see "Performing an ISSU."
- Before the system activates a feature or patch image. If the digital signature verification fails, the system will not activate the image and you will receive a digital signature verification failure message.

Restrictions and guidelines: Software upgrade

As a best practice, store the startup images in a fixed storage medium. If you store the startup images in a hot swappable storage medium, do not remove the hot swappable storage medium during the startup process.

Software upgrade is supported only on the default context.

Upgrading device software by using the boot loader method

Software upgrade tasks at a glance

To upgrade software, perform one of the following tasks:

- Upgrade the IRF fabirc:
  a. (Optional.) Preloading the BootWare image to BootWare

If a BootWare upgrade is required, you can perform this task to shorten the subsequent upgrade time. This task helps reduce upgrade problems caused by unexpected power failure. If you skip this task, the device upgrades the BootWare automatically when it upgrades the startup software images.

  b. Specifying startup images and completing the upgrade

- (Optional.) Synchronizing startup images from the master device to subordinate devices

  Perform this task when the startup images on subordinate devices are not the same version as those on the master device.

Prerequisites

1. Use the **display version** command to verify the current BootWare image version and startup software version.
2. Use the release notes for the upgrade software version to evaluate the upgrade impact on your network and verify the following items:
   - Software and hardware compatibility.
   - Version and size of the upgrade software.
   - Compatibility of the upgrade software with the current BootWare image and startup software image.
3. Use the release notes to verify whether the software images require a license. If licenses are required, register and activate licenses for each license-based software image. For more information about licensing, see "Managing licenses."
4. Use the **dir** command to verify that all IRF member devices have sufficient storage space for the upgrade images. If the storage space is not sufficient, delete unused files by using the **delete** command. For more information, see "Managing file systems."
5. Use FTP or TFTP to transfer the upgrade image file to the root directory of a file system. For more information about FTP and TFTP, see "Configuring FTP" or "Configuring TFTP." For more information about file systems, see "Managing file systems."

Preloading the BootWare image to BootWare

1. Enter system view.

   system-view

2. (Optional.) Enable BootWare image validity check.

   bootrom-update security-check enable

   By default, this feature is enabled.

   This feature examines BootWare images for file type errors, file corruption, and hardware incompatibility. As a best practice, enable it to ensure a successful upgrade.

3. Return to user view.

   quit

4. (Optional.) Back up the current BootWare image in the Normal area of BootWare.
   - Back up the image to the Backup area of BootWare:

     bootrom backup slot *slot-number-list* [**all** | **part**]

   - Back up the image to the default file system:

     bootrom read slot *slot-number-list* [**all** | **part**]

     The **bootrom read** command creates two BootWare image files on the default file system: **basicbtm.bin** for the basic segment and **extendbtm.bin** for the extended section.

Use either command to back up the BootWare image for a future version rollback or image restoration.

5. Load the upgrade BootWare image to the Normal area of BootWare.

   bootrom update file *file* **slot** *slot-number-list* [**all** | **part**]

   Specify the downloaded software image file for the *file* argument.

   NOTE:

   The system will verify the digital signature of a BootWare image before it loads it to the Normal area of BootWare. If the digital signature verification fails, the system will not load the image and you will receive a digital signature verification failure message.

   The new BootWare image takes effect at a reboot.

Specifying startup images and completing the upgrade

Perform the following steps in user view:

1. Specify main or backup startup images for all member devices.
   - Use an .ipe file:

     boot-loader file *ipe-filename* { **all** | **slot** *slot-number* } { **backup** | **main** }
   - Use .bin files:

     boot-loader file boot *filename* **system** *filename* [**feature** *filename*&<1-30>] { **all** | **slot** *slot-number* } { **backup** | **main** }

   As a best practice in a multichassis IRF fabric, specify the **all** keyword for the command. If you use the **slot** *slot-number* option to upgrade member devices one by one, version inconsistencies occur among the member devices during the upgrade.

   NOTE:

   The system will verify the digital signature of the specified images before it updates the startup image list with the specified images. If the digital signature verification fails, the system will not update the startup image list and you will receive a digital signature verification failure message.

2. Save the running configuration.

   save

   This step ensures that any configuration you have made can survive a reboot.

3. Reboot the IRF fabric.

   reboot

4. (Optional.) Verify the software image settings.

   display boot-loader [**slot** *slot-number*]

   Verify that the current software images are the same as the startup software images.

Synchronizing startup images from the master device to subordinate devices

About this task

Perform this task when the startup images on subordinate devices are not the same version as those on the master device.

This task synchronizes startup images that are running on the master device to subordinate devices. If any of the startup images does not exist or is invalid, the synchronization fails.

The startup images synchronized to subordinate devices are set as main startup images, regardless of whether the source startup images are main or backup.

Restrictions and guidelines

If an ISSU or patch installation has been performed on the master device, use the **install commit** command to update the set of main startup images on the master device before software synchronization. This command ensures startup image consistency between the master and subordinate devices.

Procedure

Perform the following steps in user view:

1. Synchronize startup images from the master to subordinate devices.
   boot-loader update { **all** | **slot** *slot-number* }
2. Reboot the subordinate devices.
   reboot slot *slot-number* [**force**]

Restoring the BootWare image

About this task

Use this task to restore the BootWare image when the BootWare image in the Normal area is corrupted or a version rollback is required.

Restrictions and guidelines

Make sure you have used the **bootrom backup** command to back up the image to the BootWare Backup area.

Procedure

Perform the following steps in user view:

1. Restore the BootWare image in the Normal area of BootWare.
   bootrom restore slot *slot-number-list* [**all** | **part**]
2. Reboot the device.
   reboot
   At startup, the system runs the new BootWare image to complete the restoration.

Display and maintenance commands for software images

Execute **display** commands in any view.

Task	Command
Display running software images and startup software images.	**display boot-loader** [**slot** *slot-number*]

Contents

Performing an ISSU

About ISSU

The In-Service Software Upgrade (ISSU) feature upgrades the NF software with a minimum amount of downtime.

Unless otherwise stated, the term "upgrade" refers to both software upgrade and downgrade in ISSU.

ISSU advantages

The NF software includes the boot image, the system image, and feature images. ISSU supports upgrading the images individually.

When you use ISSU to upgrade a feature, only the related feature images are affected. Other features can continue to provide services.

ISSU supports installing patch images to fix system bugs without a system reboot.

ISSU methods

ISSU methods are determined at software release depending on the compatibility between software versions.

ISSU supports the following upgrade types:

- **Compatible upgrade**—The new software version is compatible with the running software version. This upgrade type supports the ISSU methods in Table 1.
- **Incompatible upgrade**—The new software version is incompatible with the running software version. The two versions cannot run concurrently.

  This upgrade type supports only one upgrade method (also called incompatible upgrade). This method requires a cold reboot to upgrade both control and data planes. Incompatible upgrade disrupts service if hardware redundancy is not available.

For information about identifying the ISSU method, see "Identifying the ISSU methods."

Table 1 ISSU methods for compatible upgrade

ISSU method	Description
Incremental upgrade: • Service Upgrade • File Upgrade	Upgrades only user mode processes that have differences between the new and old software versions. Backup processes and a main/backup process switchover are required for service continuity. • **Service upgrade**—Upgrades service features. The upgrade does not affect the operation of the features that are not being upgraded. • **File upgrade**—Upgrades hidden system program files. The system can provide services during the upgrade.
Reboot	⚠ **CAUTION:** The Reboot method disrupts service if hardware redundancy is not available. As a best practice, schedule the downtime carefully to minimize the upgrade impact on the services. Reboots member devices to complete the software upgrade. While one member device is rebooting, the other member devices can provide services.

ISSU commands

ISSU includes the **`install`** and **`issu`** command sets. After you identify the recommended ISSU method, use Table 2 to choose the command set you want to use.

Table 2 Command set comparison

Item	issu commands	install commands
Upgrade types	• Compatible. • Incompatible.	Compatible.
Impact on the system	Large.	Small.
Technical skill requirements	Low. As a best practice, use this command set.	High. Administrators must have extensive system knowledge and understand the impact of each upgrade task on the network.

Restrictions and guidelines: ISSU

IMPORTANT:

- For a successful ISSU, you must remove all commands that the new version does not support and save the running configuration. To identify the feature changes between the current version and the new version, read the release notes for the device.
- To ensure correct system operation, you must remove the commands configured for features to be uninstalled and save the running configuration before uninstalling the features.

During an ISSU, follow these restrictions and guidelines:

- Do not perform any of the following tasks:
  - Reboot member devices.
  - Perform tasks that are irrelevant to the ISSU, such as modifying the configuration and displaying information.
  - Modify, delete, or rename image files.
- You cannot use both **`install`** and **`issu`** commands for the same ISSU.
- For better service continuity, strictly follow the recommended ISSU procedure. Make sure a step is completed before you proceed to the next step.
- Before executing the following commands, use the **`display system stable state`** command to verify that the system is stable:
  - **issu commands**—**`issu load`** and **`issu commit`**.
  - **install commands**—**`install activate`** and **`install deactivate`**.

  If the **System State** field displays **Stable**, the system is stable.
- You may use **`issu`** commands to upgrade all or some of the software images. If you are upgrading only some of the images, make sure the new images are compatible with the images that are not to be upgraded. The upgrade will fail if a conflict exists.
- You can install up to 32 .bin files on the device, including one boot image file, one system image file, and up to 30 feature and patch image files.
- A patch image file can be incremental or non-incremental. You can install up to 30 incremental patch image files. Because each boot, system, or feature image can have one non-incremental patch image file, you can install up to 16 non-incremental patch image files. For more

information about incremental and non-incremental patch image files, see "Upgrading software."

After an ISSU, you must log in to the device again before you can use the changed or added commands.

Prerequisites for ISSU

To perform a successful ISSU, make sure all the preparation requirements are met.

Logging in to the device through the console port

Log in to the device through the console port. If you use Telnet or SSH, you might be disconnected from the device before the ISSU is completed.

In a multiuser environment, make sure no other administrators access the device while you are performing the ISSU.

Identifying availability of ISSU and licensing requirements

Read the software release notes to identify the following items:

- Support of the device for ISSU between the current software version and the new software version.
- Licensing requirements for the upgrade software images. If the upgrade software images require licenses, make sure the device has the required licenses. For more information about license installation, see "Managing licenses."

Verifying the device operating status

Use the **display device** command to verify that all components are operating correctly.

Preparing the upgrade images

1. Use the **dir** command to verify that every file system has sufficient free storage space for the upgrade images. If the storage space is not sufficient, delete unused files by using the **delete /unreserved** *file-url* command. If the files to be deleted will be used, back up the files before deleting them. You will be unable to restore a deleted file if the **/unreserved** keyword is used. For more information, see "Managing file systems."
2. Use FTP or TFTP to transfer upgrade image files (in .bin or .ipe) to the root directory of a file system on the master device.

Identifying the ISSU methods

1. Execute the **display version comp-matrix file** command to identify the recommended ISSU methods.
    - For a compatible upgrade, check the **Upgrade Way** field to identify the recommended ISSU methods.
    - For an incompatible upgrade, check the end of command output for the **Incompatible upgrade** string.

   For more information about ISSU methods, see Table 1.

Verifying the device configuration

For a successful ISSU reboot or incompatible upgrade and the IRF fabric integrity during the ISSU, a set of features must have the same configuration as expected after the upgrade. The following are the features and the commands for you to change and verify feature settings:

- System operating mode.

  system-working-mode

  display system-working-mode
- Maximum number of ECMP routes.

  max-ecmp-num

  display max-ecmp-num
- IPv4 enhanced ECMP mode.

  ecmp mode enhanced

  display ecmp mode

For more information about system operating mode, see device management in *Fundamentals Configuration Guide*.

For more information about ECMP routes and IPv4 enhanced ECMP mode, see basic IP routing configuration in *Layer 3—IP Routing Configuration Guide*.

Verifying feature status

For service continuity during an ISSU, configure the following feature settings:

Feature	Setting requirements
GR and NSR	Enable GR or NSR for protocols including OSPF, IS-IS, BGP, and FSPF.
BFD	Disable BFD for protocols including OSPF, IS-IS, RIP, BGP, VRRP, and NQA.
Ethernet link aggregation	Use the long LACP timeout interval (the **lacp period short** command is not configured) on all member ports in dynamic aggregation groups.
IRF	Configure IRF bridge MAC persistence as follows: • **Compatible upgrade**—Configure the **irf mac-address persistent timer** or **irf mac-address persistent always** command. • **Incompatible upgrade**—Configure the **irf mac-address persistent always** command if the bridge MAC address is the MAC address of the device for which you want to execute the **issu load** command.

Determining the upgrade procedure

1. Use Table 2 to choose an upgrade command set, depending on the ISSU method.
2. Identify the hardware redundancy condition.

   ISSU can maintain service continuity only when the following conditions are met:

   - The IRF fabric has multiple members and uses the ring topology.

IMPORTANT:

If hardware redundancy is not available, service discontinuity is not avoidable during a reboot upgrade or an incompatible upgrade. Make sure you understand the impact of the upgrade on the network.

3. Choose the correct procedure from the procedures described in "Performing an ISSU by using issu commands" or "Performing an ISSU by using install commands."

Adjusting and saving the running configuration

1. Remove all commands that the new software version does not support from the running configuration. To identify all feature changes between the current version and the new version, read the release notes for the device.
2. To uninstall a feature image, remove the commands configured for the feature.
3. Use the **save** command to save the running configuration.

Performing an ISSU by using issu commands

Performing a compatible upgrade on a multichassis IRF fabricRestrictions and guidelines

Restrictions and guidelines

Upgrade a subordinate member device first. Then, upgrade the remaining member devices, including the original master.

Procedure

1. (Optional.) Configure automatic rollback:
   a. Enter system view.
      system-view
   b. Set the automatic rollback timer.
      issu rollback-timer *minutes*

      By default, the automatic rollback timer is set to 45 minutes.

      The automatic rollback timer starts when you execute the **issu run switchover** command.
   c. Return to user view.
      quit
2. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step.
3. Load the upgrade images as startup images on a subordinate member.
   - Use .bin files:

     issu load file { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> | **patch** *filename*&<1-30> } * **slot** *slot-number*&<1-9> [**reboot**]

- Use an .ipe file:

  issu load file ipe *ipe-filename* [**patch** *filename*&<1-30>] **slot** *slot-number*&<1-9> [**reboot**]

NOTE:

The software images for the device are digitally signed. The system verifies the digital signature of a software image for authenticity and integrity before it sets and loads that image as a main startup image. If the digital signature verification fails, the system will not set or load the image as a main startup image and you will receive a digital signature verification failure message.

4. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step. If the system reported an error in the previous step, verify that the system is stable and then use the **issu rollback** command to roll back the upgrade.
5. Perform an ISSU switchover.

   issu run switchover

   This command also starts the automatic rollback timer. If the timer expires, the system automatically rolls back to the original software images.
6. (Optional.) Accept the upgrade and delete the automatic rollback timer.

   issu accept

   Execute this command before the automatic rollback timer expires.
7. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step. If the system reported an error in the previous two steps, verify that the system is stable and then use the **issu rollback** command to roll back the upgrade.
8. Upgrade the remaining members to complete the ISSU.

   issu commit slot *slot-number*

   Execute this command before the automatic rollback timer expires.

△ **CAUTION:**

Repeat the previous step and this step to upgrade the remaining member devices, including the original master device. After you execute this command for one member device, wait for the member device to start up and join the IRF fabric again before you upgrade another member device. This practice ensures a successful ISSU.

9. Verify that the ISSU is finished.

   display issu state

   If the **ISSU state** field displays **Init**, the ISSU is finished.

Performing an incompatible upgrade on a multichassis IRF fabric

Restrictions and guidelines

Upgrade one or more subordinate member devices first. Then, upgrade the remaining member devices, including the original master.

Procedure

To perform an incompatible upgrade on a multichassis IRF fabric, execute the following commands in user view:

1. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step.

2. Load the upgrade images as startup images on subordinate members.

   - Use .bin files:

     issu load file { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> | **patch** *filename*&<1-30> } * **slot** *slot-number*&<1-9> [**reboot**]

   - Use an .ipe file:

     issu load file ipe *ipe-filename* [**patch** *filename*&<1-30>] **slot** *slot-number*&<1-9> [**reboot**]

   As a best practice on a ring-topology IRF fabric, specify half of the subordinate members for this command to reduce service interruption. Make sure the specified subordinate members are physically connected.

   NOTE:

   The software images for the device are digitally signed. The system verifies the digital signature of a software image for authenticity and integrity before it sets and loads that image as a main startup image. If the digital signature verification fails, the system will not set or load the image as a main startup image and you will receive a digital signature verification failure message.

3. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step. If the system reported an error in the previous step, verify that the system is stable and then use the **issu rollback** command to roll back the upgrade.

4. Perform an ISSU switchover to complete the ISSU process.

   issu run switchover

   This command upgrades the remaining members.

5. Verify that the ISSU is finished.

   display issu state

   If the **ISSU state** field displays **Init**, the ISSU is finished.

Performing an incremental upgrade on a single-chassis IRF fabric

To perform an incremental upgrade on a single-chassis IRF fabric, execute the following commands in user view:

1. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step.

2. Load the upgrade images as startup images.

- Use .bin files:

  issu load file { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> | **patch** *filename*&<1-30> } * **slot** *slot-number* [**reboot**]

- Use an .ipe file:

  issu load file ipe *ipe-filename* [**patch** *filename*&<1-30>] **slot** *slot-number* [**reboot**]

NOTE:

The software images for the device are digitally signed. The system verifies the digital signature of a software image for authenticity and integrity before it sets and loads that image as a main startup image. If the digital signature verification fails, the system will not set or load the image as a main startup image and you will receive a digital signature verification failure message.

3. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step. If the system reported an error in the previous step, verify that the system is stable and then use the **issu rollback** command to roll back the upgrade.

4. Complete the ISSU process.

   issu commit slot *slot-number*

CAUTION:

The ISSU process cannot be rolled back automatically or manually after you execute this command. When the ISSU commit operation is completed, the ISSU status changes to Init.

5. Verify that the ISSU is finished.

   display issu state

   If the **ISSU state** field displays **Init**, the ISSU is finished.

Performing a reboot or incompatible upgrade on a single-chassis IRF fabric

1. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step.

2. Load the parent device's upgrade images as startup images on subordinate members.

   - Use .bin files:

     issu load file { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> | **patch** *filename*&<1-30> } * **slot** *slot-number* [**reboot**]

   - Use an .ipe file:

     issu load file ipe *ipe-filename* [**patch** *filename*&<1-30>] **slot** *slot-number* [**reboot**]

NOTE:

The software images for the device are digitally signed. The system verifies the digital signature of a software image for authenticity and integrity before it sets and loads that image as a main startup image. If the digital signature verification fails, the system will not set or load the image as a main startup image and you will receive a digital signature verification failure message.

3. Verify that the ISSU is finished.

   ```
   display issu state
   ```

   If the **ISSU state** field displays **Init**, the ISSU is finished.

Performing an ISSU by using install commands

ISSU tasks at a glance

1. (Optional.) Decompressing an .ipe file
2. Installing and upgrading software images
3. (Optional.) Deactivating software images
4. (Optional.) Rolling back the running software images
5. (Optional.) Aborting a software activate or deactivate operation
6. (Optional.) Verifying software images
7. Committing software changes
8. (Optional.) Deleting inactive software images

Decompressing an .ipe file

1. (Optional.) Identify images that are included in the .ipe file.

   ```
   display install ipe-info
   ```

2. Decompress the .ipe file.

   ```
   install add ipe-filename filesystem
   ```

Installing and upgrading software images

About this task

Use this task to install new features and patch images or upgrade the running boot, system, or feature images.

Software image installation and upgrade methods

Use one of the following methods to install or upgrade software images:

- **Slot by slot**—Activate all the images on one slot, and then move to the next slot.
- **Image by image**—Activate one image on all slots before activating another image.

Restrictions and guidelines

To install an image, you must begin with the master device. To upgrade an image, you must begin with a subordinate device.

The activate operation for an incremental upgrade or patch images only updates the current software image list. For the image changes to take effect after a reboot, you must perform a commit operation to update the main startup image list.

Installing or upgrading boot, system, and feature images

To install or upgrade boot, system, and feature images, execute the following commands in user view:

1. Verify that the system is stable.

display system stable state

The system is stable if the **System State** field displays **Stable**. For a successful ISSU, you must make sure the system is stable before you proceed to the next step.

2. (Optional.) Identify the recommended ISSU method and the possible impact of the upgrade.

   install activate { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> } * **slot** *slot-number* **test**

3. Activate images.

   install activate { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> } * **slot** *slot-number*

Installing patch images

To install patch images, execute the following commands in user view:

1. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful installation, you must make sure the system is stable before you proceed to the next step.

2. Activate patch images.

   install activate patch *filename* { **all** | **slot** *slot-number* }

   The **install activate patch** *filename* **all** command installs the specified patch images on all hardware and the images can survive a reboot. You do not need to execute the **install commit** command for the installation.

   You can specify only one patch image file for the command at a time. However, you can execute the command multiple times to activate multiple patch image files.

Deactivating software images

Restrictions and guidelines

You can deactivate only feature and patch images.

The deactivate operation only removes images from the current software image list. For the image changes to take effect after a reboot, you must perform a commit operation to remove the images from the main startup image list.

Deactivated images are still stored on the storage medium. To permanently delete the images, execute the **install remove** command. For more information, see "Deleting inactive software images."

Deactivating feature images

To deactivate feature images, execute the following commands in user view:

1. Verify that the system is stable.

   display system stable state

   The system is stable if the **System State** field displays **Stable**. For a successful deactivate operation, you must make sure the system is stable before you proceed to the next step.

2. Deactivate feature images.

   install deactivate feature *filename*&<1-30> **slot** *slot-number*

Deactivating patch images

To deactivate patch images, execute the following commands in user view:

1. Verify that the system is stable.

   display system stable state

The system is stable if the **System State** field displays **Stable**. For a successful deactivate operation, you must make sure the system is stable before you proceed to the next step.

2. Deactivate patch images.

install deactivate patch *filename* { **all** | **slot** *slot-number* }

The **install deactivate patch** *filename* **all** command deactivates the specified patch images on all hardware and the image changes can survive a reboot. You do not need to execute the **install commit** command for the deactivation.

You can specify only one patch image file for the command at a time. However, you can execute the command multiple times to deactivate multiple patch image files.

Rolling back the running software images

About this task

During an incremental upgrade, the system creates a rollback point for each activate or deactivate operation of a boot, system, or feature image. The system can maintain a maximum of 50 rollback points. If this limit has been reached when a rollback point is created, the system removes the earliest rollback point. You can roll back the software to any of the rollback points.

During a reboot upgrade, the system does not create rollback points. After a reboot upgrade, you can roll back the software only to the status before any activate or deactivate operations were performed.

Restrictions and guidelines

You can perform this task only before committing software changes. A commit operation deletes all rollback points.

For an incremental upgrade rollback to take effect after a reboot, you must perform a commit operation to update the main startup image list.

Procedure

To roll back the running software images, execute the following commands in user view:

1. (Optional.) Display available rollback points.

display install rollback

2. Roll back the software.

install rollback to { *point-id* | **original** }

Aborting a software activate or deactivate operation

About this task

While the system is activating or deactivating a software image for a service upgrade or file upgrade, you can abort the activate or deactivate operation. After an operation is aborted, the system runs with the software images that it was running with before the operation.

Procedure

To abort a software activate or deactivate operation, use one of the following methods:

- Execute the **install abort** [*job-id*] command in user view.
- Press **Ctrl+C**.

Committing software changes

About this task

When you activate or deactivate images for an incremental upgrade, or activate or deactivate patch images, the main startup image list does not update with the changes. The software changes are lost at reboot. For the changes to take effect after a reboot, you must commit the changes.

Procedure

To commit software changes, execute the following command in user view:

install commit

Verifying software images

About this task

Perform this task to verify the following items:

- **Integrity**—Verify that the boot, system, and feature images are integral.
- **Consistency**—Verify that the same active images are running across the entire system.
- **Software commit status**—Verify that the active images are committed as needed.

Procedure

To verify software images, execute the following commands in user view:

1. Verify software images.
   install verify
2. Activate or deactivate images as required.
   install { **activate** | **deactivate** }
3. Commit the software changes.
   install commit

Deleting inactive software images

About this task

After completing an ISSU, you can use this task to delete old image files permanently.

Restrictions and guidelines

This task deletes image files permanently. You cannot use the **install rollback to** command to revert the operation, or use the **install abort** command to abort the operation.

Procedure

To delete inactive software image files, execute the following command in user view:

install remove [**slot** *slot-number*] { *filename* | **inactive** }

Terminating the ongoing ISSU process forcibly

About this task

Use this feature to terminate the ISSU process if one of the following exceptions occurs and you cannot perform an upgrade or rollback:

- The ISSU status is not **Init** but the upgrade has stopped.

- The ISSU status is **Init** but the upgrade has not completed.

Procedure

1. Enter system view.
   system-view
2. Terminate the ongoing ISSU process forcibly.
   issu quit

Display and maintenance commands for ISSU

△ **CAUTION:**

The **reset install rollback oldest** command clears the specified rollback point and all rollback points earlier than the specified rollback point. You will be unable to roll back the configuration to the status when any of these rollback points was created.

Unless otherwise stated, the **display** and **reset** commands can be used during an ISSU, regardless of whether the **install** or **issu** commands are used.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display active software images.	**display install active** [**slot** *slot-number*] [**verbose**]
Display backup startup software images.	**display install backup** [**slot** *slot-number*] [**verbose**]
Display main startup software images.	**display install committed** [**slot** *slot-number*] [**verbose**]
Display inactive software images in the root directories of file systems.	**display install inactive** [**slot** *slot-number*] [**verbose**]
Display the software images included in an .ipe file.	**display install ipe-info** *ipe-filename* N/A
Display ongoing ISSU activate, deactivate, and rollback operations.	**display install job** N/A
Display ISSU log entries.	**display install log** [*log-id*] [**verbose**] N/A
Display software image file information.	**display install package** { *filename* | **all** } [**verbose**] N/A
Display rollback point information. (The system does not record rollback points during an ISSU that uses **issu** commands.)	**display install rollback** [*point-id*]
Display the software image file that includes a specific component or file.	**display install which** { **component** *name* | **file** *filename* } [**slot** *slot-number*] N/A
Display automatic rollback timer	**display issu rollback-timer**

Task	Command
information.	
Display ISSU status information. (This command applies only to an ISSU that uses **issu** commands.)	**display issu state**
Display the ISSU methods.	**display version comp-matrix file** { **boot** *filename* | **system** *filename* | **feature** *filename*&<1-30> } * **display version comp-matrix file ipe** *ipe-filename*
Clear ISSU log entries.	**reset install log-history oldest** *log-number*
Clear ISSU rollback points.	**reset install rollback oldest** *point-id*

Examples of using issu commands for ISSU

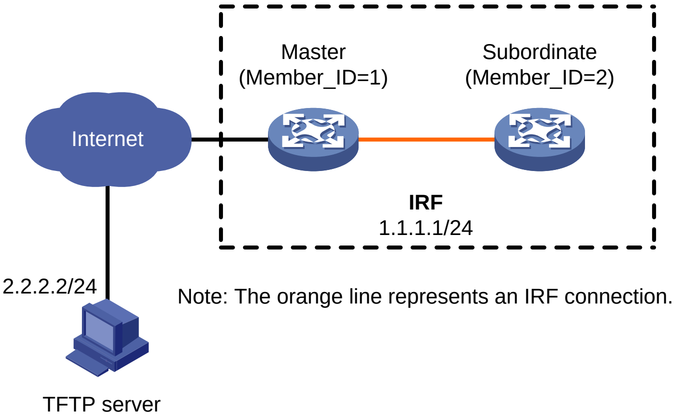
Example: Upgrading the system software to a compatible version

Network configuration

As shown in Figure 1, the IRF fabric has two members.

Upgrade the boot, system, and feature images from T0001015 to T0001016. The two versions are compatible.

Figure 1 Network diagram



Procedure

# Download the upgrade image files from the TFTP server.

```
<Sysname> tftp 2.2.2.2 get boot-t0001016.bin
  % Total     % Received % Xferd  Average Speed   Time    Time     Time  Current
                                  Dload  Upload   Total   Spent    Left  Speed
100   6078464  100  6078464    0     0    764      0 --:--:-- --:--:-- --:--:--    810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get system-t0001016.bin
  % Total     % Received % Xferd  Average Speed   Time    Time     Time  Current
```

```
                                 Dload  Upload   Total   Spent    Left  Speed
100   97750016  100  97750016     0     0    764      0 --:--:-- --:--:-- --:--:--   810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get feature-t0001016.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   1008640  100  1008640     0     0    764      0 --:--:-- --:--:-- --:--:--   810
Writing file...Done.
```

# Display active software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Active packages on slot 2:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
```

# Identify the recommended ISSU methods.

```
<Sysname> display version comp-matrix file boot flash:/boot-t0001016.bin system
flash:/system-t0001016.bin feature flash:/feature-t0001016.bin
Verifying the file flash:/boot-t0001016.bin on slot 1......................Done.
Verifying the file flash:/system-t0001016.bin on slot
1................ ................Done.
Verifying the file flash:/feature-t0001016.bin on slot 1................Done.
Identifying the upgrade methods....Done.

  Slot                        Upgrade Way
  1                           Reboot
  2                           Reboot
```

The output shows that reboot upgrades are recommended.

# Save the running configuration.

```
<Sysname> save
```

# Upgrade the system software on the subordinate member.

```
<Sysname> issu load file boot flash:/boot-t0001016.bin system flash:/system-t0001016.bin
feature flash:/feature-t0001016.bin slot 2
This operation will delete the rollback point information for the previous upgrade and
maybe get unsaved configuration lost. Continue? [Y/N]:y
Copying file flash:/boot-t0001016.bin to slot2#flash:/boot-t0001016.bin......Done.
Copying file flash:/system-t0001016.bin to
slot2#flash:/system-t0001016.bin.......................Done.
Copying file flash:/feature-t0001016.bin to
slot2#flash:/feature-t0001016.bin......Done.
Verifying the file flash:/boot-t0001016.bin on slot 2...Done.
Verifying the file flash:/system-t0001016.bin on slot 2...........Done.
Verifying the file flash:/feature-t0001016.bin on slot 2...Done.
Identifying the upgrade methods….Done.
Upgrade summary according to following table:
```

```
flash:/boot-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/system-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/feature-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

  Slot                        Upgrade Way
  2                           Reboot
Upgrading software images to compatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait...Done.
```

# Verify that the ISSU is in a stable state.

```
<Sysname> display issu state
ISSU state: Loaded
Compatibility: Compatible
Work state: Normal
Upgrade method: Card by card
Upgraded slot:
  slot 2
Current upgrading slot: None
Previous version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Previous software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Upgrade version list:
  boot: 7.1.070, Test 0001016
  system: 7.1.070, Test 0001016
  feature: 7.1.070, Test 0001016
Upgrade software images:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

The **Loaded** state is a stable state, which indicates that the system is waiting for a master/subordinate switchover. Slot 2 has completed the upgrade, and slot 1 has not.

# Perform a master/subordinate switchover.

```
<Sysname> issu run switchover
Upgrade summary according to following table:
```

```
flash:/boot-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/system-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/feature-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

  Slot                        Switchover Way
  1                           Master subordinate switchover
Upgrading software images to compatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait.........
```

# Verify that the ISSU is in a stable state.

```
<Sysname> display issu state
ISSU state: Switchover
Compatibility: Compatible
Work state: Normal
Upgrade method: Card by card
Upgraded slot:
  slot 2
Current upgrading slot: None
Previous version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Previous software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Upgrade version list:
  boot: 7.1.070, Test 0001016
  system: 7.1.070, Test 0001016
  feature: 7.1.070, Test 0001016
Upgrade software images:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

The **Switchover** state is a stable state, which indicates that the system has completed the master/subordinate switchover and is waiting for a commit operation to upgrade slot 1.

# Upgrade the system software on the original master.

```
<Sysname> issu commit slot 1
Upgrade summary according to following table:

flash:/boot-t0001016.bin
```

```
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/system-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/feature-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

  Slot                        Upgrade Way
  1                           Reboot
Upgrading software images to compatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait...Done.
```

Verifying the upgrade

# Verify that the ISSU is finished.

```
<Sysname> display issu state
ISSU state: Init
Compatibility: Unknown
Work state: Normal
Upgrade method: Card by card
Upgraded slot: None
Current upgrading slot: None
Current version list:
  boot: 7.1.070, Test 0001016
  system: 7.1.070, Test 0001016
  feature: 7.1.070, Test 0001016
Current software images:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

The **ISSU state** field displays **Init**, which indicates that the ISSU is finished.

# Verify that both members are running the new software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
Active packages on slot 2:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

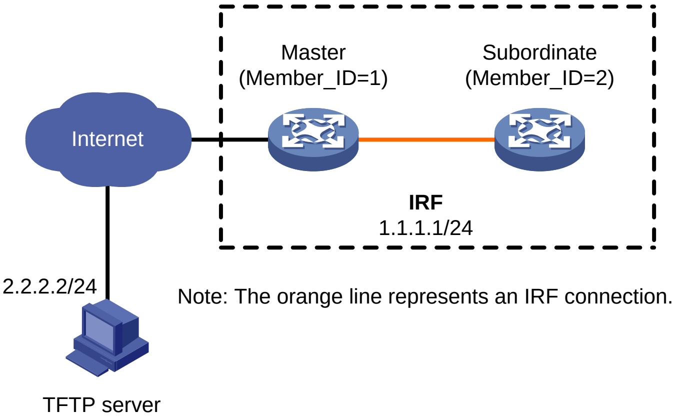
Example: Upgrading the system software to an incompatible version

Network configuration

As shown in Figure 2, the IRF fabric has two members.

Upgrade the boot, system, and feature images from T0001015 to T0001017, which is an incompatible version.

Figure 2 Network diagram



Note: The orange line represents an IRF connection.

Procedure

# Download the upgrade image files from the TFTP server.

```
<Sysname> tftp 2.2.2.2 get boot-t0001017.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   6078464  100  6078464     0     0    764      0 --:--:-- --:--:-- --:--:--    810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get system-t0001017.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   97750016  100  97750016     0     0    764      0 --:--:-- --:--:-- --:--:--    810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get feature-t0001017.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   1008640  100  1008640     0     0    764      0 --:--:-- --:--:-- --:--:--    810
Writing file...Done.
```

# Display active software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Active packages on slot 2:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
```

# Identify the recommended ISSU methods.

```
<Sysname> display version comp-matrix file boot flash:/boot-t0001017.bin system
flash:/system-t0001017.bin feature flash:/feature-t0001017.bin
Verifying the file flash:/boot-t0001017.bin on slot 1.......................Done.
Verifying the file flash:/system-t0001017.bin on slot
1................. .................Done.
Verifying the file flash:/feature-t0001017.bin on slot 1..................Done.
Identifying the upgrade methods....Done.

Incompatible upgrade.
```

The output shows that an incompatible upgrade is recommended.

# Save the running configuration.

```
<Sysname> save
```

# Upgrade the system software on the subordinate member.

```
<Sysname> issu load file boot flash:/boot-t0001017.bin system flash:/system-t0001017.bin
feature flash:/feature-t0001017.bin slot 2
This operation will delete the rollback point information for the previous upgrade and
maybe get unsaved configuration lost. Continue? [Y/N]:y
Copying file flash:/boot-t0001017.bin to slot2#flash:/boot-t0001017.bin......Done.
Copying file flash:/system-t0001017.bin to
slot2#flash:/system-t0001017.bin........................Done.
Copying file flash:/feature-t0001017.bin to
slot2#flash:/feature-t0001017.bin......Done.
Verifying the file flash:/boot-t0001017.bin on slot 2...Done.
Verifying the file flash:/system-t0001017.bin on slot 2...........Done.
Verifying the file flash:/feature-t0001017.bin on slot 2...Done.
Identifying the upgrade methods….Done.
Upgrade summary according to following table:

flash:/boot-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

flash:/system-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

flash:/feature-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

  Slot                        Upgrade Way
  2                           Reboot
Upgrading software images to incompatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait...Done.
```

# Verify that the ISSU is in a stable state.

```
<Sysname> display issu state
```

```
ISSU state: Loaded
Compatibility: Incompatible
Work state: Normal
Upgrade method: Card by card
Upgraded slot:
  slot 2
Current upgrading slot: None
Previous version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Previous software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Upgrade version list:
  boot: 7.1.070, Test 0001017
  system: 7.1.070, Test 0001017
  feature: 7.1.070, Test 0001017
Upgrade software images:
  flash:/boot-t0001017.bin
  flash:/system-t0001017.bin
  flash:/feature-t0001017.bin
```

The **Loaded** state is a stable state, which indicates that the system is waiting for a master/subordinate switchover. Slot 2 has completed the upgrade, and slot 1 has not.

# Perform a master/subordinate switchover.

```
<Sysname> issu run switchover
Upgrade summary according to following table:

flash:/boot-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

flash:/system-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

flash:/feature-t0001017.bin
  Running Version             New Version
  Test 0001015                Test 0001017

  Slot                        Switchover Way
  1                           Master subordinate switchover
Upgrading software images to incompatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait...Done.
```

Verifying the upgrade

# Verify that the ISSU is finished.

```
<Sysname> display issu state
```

```
ISSU state: Init
Compatibility: Unknown
Work state: Normal
Upgrade method: Card by card
Upgraded slot: None
Current upgrading slot: None
Current version list:
  boot: 7.1.070, Test 0001017
  system: 7.1.070, Test 0001017
  feature: 7.1.070, Test 0001017
Current software images:
  flash:/boot-t0001017.bin
  flash:/system-t0001017.bin
  flash:/feature-t0001017.bin
```

The **ISSU state** field displays **Init**, which indicates that the ISSU is finished.

# Verify that both members are running the new software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001017.bin
  flash:/system-t0001017.bin
  flash:/feature-t0001017.bin
Active packages on slot 2:
  flash:/boot-t0001017.bin
  flash:/system-t0001017.bin
  flash:/feature-t0001017.bin
```

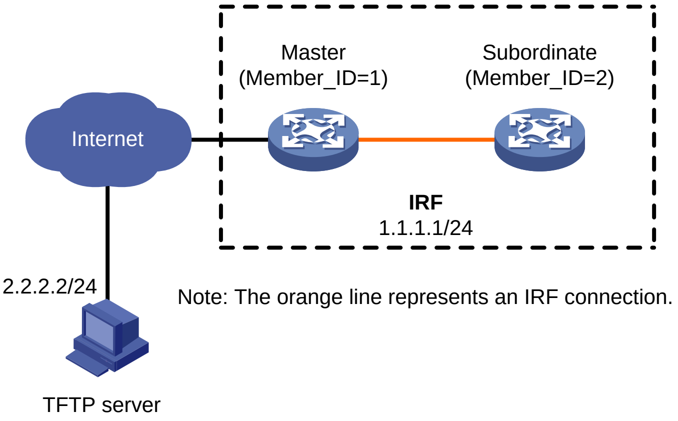
Example: Rolling back the system software

Network configuration

As shown in Figure 3, the IRF fabric has two members.

Roll back the boot, system, and feature images from T0001016 to T0001015 after upgrading them from T0001015 to T0001016. T0001016 and T0001015 are compatible.

Figure 3 Network diagram



Procedure

# Download the upgrade image files from the TFTP server.

```
<Sysname> tftp 2.2.2.2 get boot-t0001016.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   6078464  100  6078464     0     0    764      0 --:--:-- --:--:-- --:--:--   810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get system-t0001016.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   97750016  100  97750016     0     0    764      0 --:--:-- --:--:-- --:--:--   810
Writing file...Done.
<Sysname> tftp 2.2.2.2 get feature-t0001016.bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   1008640  100  1008640     0     0    764      0 --:--:-- --:--:-- --:--:--   810
Writing file...Done.
```

# Display active software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Active packages on slot 2:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
```

# Identify the recommended ISSU methods.

```
<Sysname> display version comp-matrix file boot flash:/boot-t0001016.bin system
flash:/system-t0001016.bin feature flash:/feature-t0001016.bin
Verifying the file flash:/all.ipe on slot 1................ ................Done.
Identifying the upgrade methods....Done.

  Slot                        Upgrade Way
  1                           Reboot
  2                           Reboot
```

The output shows that reboot upgrades are recommended.

# Save the running configuration.

```
<Sysname> save
```

# Upgrade the system software on the subordinate member.

```
<Sysname> issu load file boot flash:/boot-t0001016.bin system flash:/system-t0001016.bin
feature flash:/feature-t0001016.bin slot 2
This operation will delete the rollback point information for the previous upgrade and
maybe get unsaved configuration lost. Continue? [Y/N]:y
Copying file flash:/boot-t0001016.bin to slot2#flash:/boot-t0001016.bin......Done.
Copying file flash:/system-t0001016.bin to
slot2#flash:/system-t0001016.bin........................Done.
```

```
Copying file flash:/feature-t0001016.bin to
slot2#flash:/feature-t0001016.bin......Done.
Verifying the file flash:/boot-t0001016.bin on slot 2...Done.
Verifying the file flash:/system-t0001016.bin on slot 2............Done.
Verifying the file flash:/feature-t0001016.bin on slot 2...Done.
Identifying the upgrade methods….Done.
Upgrade summary according to following table:

flash:/boot-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/system-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/feature-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

  Slot                        Upgrade Way
  2                           Reboot
Upgrading software images to compatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait...Done.
```

# Verify that the ISSU is in a stable state.

```
<Sysname> display issu state
ISSU state: Loaded
Compatibility: Compatible
Work state: Normal
Upgrade method: Card by card
Upgraded slot:
  slot 2
Current upgrading slot: None
Previous version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Previous software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Upgrade version list:
  boot: 7.1.070, Test 0001016
  system: 7.1.070, Test 0001016
  feature: 7.1.070, Test 0001016
Upgrade software images:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

The **Loaded** state is a stable state, which indicates that the system is waiting for a master/subordinate switchover. Slot 2 has completed the upgrade, and slot 1 has not.

# Perform a master/subordinate switchover.

```
<Sysname> issu run switchover
Upgrade summary according to following table:

flash:/boot-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/system-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

flash:/feature-t0001016.bin
  Running Version             New Version
  Test 0001015                Test 0001016

  Slot                        Switchover Way
  1                           Master subordinate switchover
Upgrading software images to compatible versions. Continue? [Y/N]:y
This operation might take several minutes, please wait.........
```

# Verify that the ISSU is in a stable state.

```
<Sysname> display issu state
ISSU state: Switchover
Compatibility: Compatible
Work state: Normal
Upgrade method: Card by card
Upgraded slot:
  slot 2
Current upgrading slot: None
Previous version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Previous software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Upgrade version list:
  boot: 7.1.070, Test 0001016
  system: 7.1.070, Test 0001016
  feature: 7.1.070, Test 0001016
Upgrade software images:
  flash:/boot-t0001016.bin
  flash:/system-t0001016.bin
  flash:/feature-t0001016.bin
```

The **Switchover** state is a stable state, which indicates that the system has completed the master/subordinate switchover and is waiting for a commit operation to upgrade slot 1.

# Roll back the software images to T0001015.

```
<Sysname> issu rollback
This command will quit the ISSU process and roll back to the previous version. Continue?
[Y/N]:Y
```

# Verify that the rollback is finished.

```
<Sysname> display issu state
ISSU state: Init
Compatibility: Unknown
Work state: Normal
Upgrade method: Card by card
Upgraded slot: None
Current upgrading slot: None
Current version list:
  boot: 7.1.070, Test 0001015
  system: 7.1.070, Test 0001015
  feature: 7.1.070, Test 0001015
Current software images:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
```

The **ISSU state** field displays **Init**, which indicates that the rollback is finished.

# Verify that both members are running the old software images.

```
<Sysname> display install active
Active packages on slot 1:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
Active packages on slot 2:
  flash:/boot-t0001015.bin
  flash:/system-t0001015.bin
  flash:/feature-t0001015.bin
```

Contents

Using automatic configuration

About automatic configuration

With the automatic configuration feature, the device can automatically obtain a set of configuration settings at startup. You only need to save the configuration file on a specific storage medium. This feature simplifies network configuration and maintenance.

The device supports only USB-based automatic configuration. USB-based automatic configuration applies to the following scenarios:

- Small networks where the devices reside near to each other and no host can be used as a file server.
- Large networks where only a few devices require automatic configuration or configuration update.

Restrictions and guidelines: Automatic configuration

USB disk-based automatic configuration supports configuring the default context but does not support configuring non-default contexts.

The device performs USB disk-based automatic configuration at startup only if it does not have a configuration file.

Using USB-based automatic configuration

About USB-based automatic configuration

USB-based automatic configuration enables the device to obtain a configuration file from a connected USB disk at startup.

After obtaining a configuration file, the device compares the file with its main startup configuration file. If the two files have the same settings, the device loads its main startup configuration file. If the two files have different settings, the device performs the following operations:

1. Identifies whether its main startup configuration file is using the same name as the obtained configuration file.
   - If yes, the device renames its main startup configuration file by adding **_bak** to the base name of the file, and copies the obtained configuration file.
   - If not, the system uses the obtained configuration file to overwrite its main startup configuration file.
2. Loads the obtained configuration file.
   - If all commands in the obtained configuration file are successfully loaded, the device sets the obtained configuration file as the main startup configuration file.
   - If a command in the obtained configuration file fails, the device removes all loaded settings and searches for a local configuration file.
     - If a configuration file is found, the device loads the configuration file.
     - If no configuration file is found, the device finishes the automatic configuration process without loading any configurations.

Preparing the USB disk for automatic configuration

1. Prepare a USB disk that has only one partition.
2. Display the serial number of the device.

   ```
   display device manuinfo
   ```

   For more information about this command, see *Fundamentals Command Reference*.
3. Create a configuration file named *Device serial number***.cfg** or **autodeploy.cfg**, and save the file to the root directory of the file system on the USB disk.

   If a configuration file named *Device serial number***.cfg** coexists with configuration file **autodeploy.cfg**, configuration file *Device serial number***.cfg** is used.

Configuring and using USB-based automatic configuration

1. Enable USB-based automatic configuration on the device:
   a. Enter system view.

      ```
      system-view
      ```

   b. Enable USB-based automatic configuration.

      ```
      autodeploy udisk enable
      ```

      By default, USB-based automatic configuration is enabled.

   c. Save the running configuration.

      ```
      save
      ```

      A device reboot is required for USB-based automatic configuration. Save the running configuration to ensure that the USB-based automatic configuration feature takes effect after a reboot.
2. If the IRF fabric has two or more member devices, shut down the IRF physical interfaces by using the **shutdown** command to split the IRF fabric.
3. Connect the USB disk to the USB1 interface on the master device of the original IRF fabric.

   The USB disk will be identified as usba0.
4. Reboot the device and observe the LEDs of the device.

   If the device finds no configuration files to load at startup, it performs the following operations:

   a. Copies the configuration file on the USB disk.

   b. Specifies the file as the startup configuration file.

   c. Reboots to load the configuration file.

   - If the SYS LED flashes green quickly for 5 seconds, the automatic configuration succeeded. Proceed to step 5.
   - If the SYS LED flashes yellow quickly for 10 seconds, the automatic configuration failed. Display the log file named *Fully qualified configuration file name***.log** in the USB disk root directory to locate and resolve the problem.

   For more information about the LEDs, see the installation guide.
5. If the automatic configuration succeeded, use the **display current-configuration** command to verify that the configuration file has been loaded correctly.
6. Rebuild the IRF fabric.

   The subordinate members automatically synchronize their configurations with the master at startup.

Contents

Using Tcl

About Tcl

NF provides a built-in tool command language (Tcl) interpreter. From user view, you can use the **`tclsh`** command to enter Tcl configuration view to execute the following commands:

- Tcl 8.5 commands.
- NF commands.

The Tcl configuration view is equivalent to the user view. You can use NF commands in Tcl configuration view in the same way they are used in user view.

Restrictions and guidelines: Tcl

To return from a subview under Tcl configuration view to the upper-level view, use the **`quit`** command.

To return from a subview under Tcl configuration view to the Tcl configuration view, press **Ctrl+Z**.

Using Tcl commands to configure the device

Restrictions and guidelines

When you use Tcl to configure the device, follow these restrictions and guidelines:

- You can apply Tcl environment variables to NF commands.
- No online help information is provided for Tcl commands.
- You cannot press **Tab** to complete an abbreviated Tcl command.
- Make sure the Tcl commands can be executed correctly.
- As a best practice, log in through Telnet or SSH. You cannot stop Tcl commands by using a shortcut key or a CLI command. If a problem occurs when the Tcl commands are being executed, you can terminate the process by closing the connection if you logged in through Telnet or SSH. If you logged in from the console port, you must perform one of the following tasks:
    - Restart the device.
    - Log in to the device by using a different method, and use the **`free line`** command to release the console line. For more information about the command, see *Fundamentals Command Reference*.
- You can press **Ctrl+D** to abort Tcl command **`read stdin`**.

Procedure

1. Enter Tcl configuration view from user view.
   `tclsh`
2. Execute a Tcl command.
   `Tcl command`
3. Return from Tcl configuration view to user view.

- **tclquit**
- **quit**

Executing NF commands in Tcl configuration view

About executing NF commands in Tcl configuration view

To execute a NF command in Tcl configuration view, use one of the following methods:

- Enter the NF command directly. If a Tcl command uses the same command string as the NF command, the Tcl command is executed.
- Prefix the NF command with the **cli** keyword. If a Tcl command uses the same command string as the NF command, the NF command is executed.

Restrictions and guidelines

Follow these restrictions and guidelines when you execute NF commands in Tcl configuration view:

- To specify a string enclosed in quotation marks (") or braces ({ and }), you must use the escape character (\) before the quotation marks or braces. For example, to specify **"a"** as the description for an interface, you must enter **description \"a\"**. If you enter **description "a"**, the description is **a**.
- For NF commands, you can enter **?** to obtain online help or press **Tab** to complete an abbreviated command. For more information, see "Using the CLI."
- The **cli** command is a Tcl command, so you cannot enter **?** to obtain online help or press **Tab** to complete an abbreviated command.
- Successfully executed NF commands are saved to command history buffers. You can use the upper arrow or lower arrow key to obtain executed commands.
- To execute multiple NF commands in one operation:
  - Enter multiple NF commands separated by semi-colons to execute the commands in the order they are entered. For example, **ospf 100**; **area 0**.
  - Specify multiple NF commands for the **cli** command, quote them, and separate them by a space and a semicolon. For example, **cli** "**ospf 100** ; **area 0**".
  - Specify one NF command for each **cli** command and separate them by a space and a semicolon. For example, **cli ospf 100** ; **cli area 0**.

Procedure

1. Enter Tcl configuration view

   tclsh

2. Execute NF commands.
   - Execute NF commands directly.

     Command
   - Execute NF commands by using the **cli** command.

     cli *command*
3. Return from Tcl configuration view to user view.
   - **tclquit**
   - **quit**

Contents

Using Python

About Python

NF provides a built-in Python interpreter. You can use Python to perform the following tasks:

- Execute Python scripts to implement automatic device configuration.
- Enter Python shell to configure the device by using the following items:
  - Python 2.7 commands.
  - Python 2.7 standard API.
  - Extended API. For more information about the extended API, see "NF extended Python API."

Executing a Python script

To execute a Python script, use the following command in user view:

python *filename*

Entering the Python shell

To enter the Python shell from user view, execute the following command:

python

Importing and using the extended Python API

To use the extended Python API, you must first import the API to Python.

Importing the entire extended API and using the API

Procedure

1. Enter the Python shell from user view.
   python
2. Import the entire extended API.
   import platformtools
3. Execute an extended API function.
   platformtools.*api*

Example

# Use extended API function **Transfer** to download the **test.cfg** file from TFTP server 192.168.1.26.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.Transfer('tftp', '192.168.1.26', 'test.cfg', 'flash:/test.cfg',
user='', password='')
<platformtools.Transfer object at 0xb7eab0e0>
```

Importing an extended API function and using the function

Procedure

1. Enter the Python shell from user view.
   `python`
2. Import an extended API function.
   `from platformtools import` *`api-name`*
3. Execute an extended API function.
   `api-function`

Example

# Use extended API function **`Transfer`** to download the **test.cfg** file from TFTP server 192.168.1.26.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from platformtools import Transfer
>>> Transfer('tftp', '192.168.1.26', 'test.cfg', 'flash:/test.cfg', user='',
password='')
<platformtools.Transfer object at 0xb7e5e0e0>
```

Exiting the Python shell

To exit the Python shell, execute the following command in the Python shell.

`exit()`

NF extended Python API

The NF extended Python API is compatible with the Python syntax.

CLI

Use **`CLI`** to execute NF CLI commands and create CLI objects.

Syntax

`CLI`(*`command`*=‘’, *`do_print`*=True)

Parameters

`command`: Specifies the commands to be executed. To enter multiple commands, use a space and a semicolon (;) as the delimiter. To enter a command in a view other than user view, you must first enter the commands used to enter the view. For example, you must enter **`’system-view ;local-user test class manage’`** to execute the **`local-user test class manage`** command.

`do_print`: Specifies whether to output the execution result:

- **True**—Outputs the execution result. This value is the default.
- **False**—Does not output the execution result.

Usage guidelines

This API function supports only NF commands. It does not support Linux, Python, or Tcl commands.

Returns

CLI objects

Examples

# Add a local user named **test**.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.CLI('system-view ;local-user test class manage')
```

Sample output

```
<Sysname> system-view
System View: return to User View with Ctrl+Z.
[Sysname] local-user test class manage
New local user added.
<platformtools.CLI object at 0xb7f680a0>
```

get_error

Use **`get_error`** to get the error information from the download operation.

Syntax

`Transfer.get_error()`

Returns

Error information (if there is no error information, the return is null)

Examples

# Download file **test.cfg** from TFTP server 1.1.1.1 and get the error information from the operation.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> c = platformtools.Transfer('tftp', '1.1.1.1', 'test.cfg', 'flash:/test.cfg', user='',
password='')
>>> c.get_error()
```

Sample output

```
"Timeout was reached”
```

get_output

Use **`get_output`** to get the output from executed commands.

Syntax

`CLI.get_output()`

Returns

Output from executed commands

Examples

# Add a local user and get the output from the command.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> c = platformtools.CLI('system-view ;local-user test class manage', False)
>>> c.get_output()
```

Sample output

```
['<Sysname>system-view', 'System View: return to User View with Ctrl+Z.',
'[Sysname]local-user test class manage', 'New local user added.']
```

get_self_slot

Use **`get_self_slot`** to get the member ID of the master device.

Syntax

`get_self_slot()`

Returns

A list object in the format of `[`-1,*slot-number*`]`. The *slot-number* indicates the member ID of the master device.

Examples

# Get the member ID of the master device.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.get_self_slot()
```

Sample output

```
[-1,1]
```

get_slot_info

Use **`get_slot_info`** to get information about a member device.

Syntax

`get_slot_info()`

Returns

A dictionary object in the format of {'Slot': *slot-number*, 'Status': '*status*', 'Chassis': *chassis-number*, 'Role': '*role*', 'Cpu': *CPU-number* }. The *slot-number* argument indicates the member ID of the device. The *status* argument indicates the status of the member device. The *chassis-number* and *CPU-number* arguments are fixed at 0. The *role* argument indicates the role of the member device.

Examples

# Get information about the device, a card, or a member device.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.get_slot_info(1)
```

Sample output

```
{'Slot': 1, 'Status': 'Normal', 'Chassis': 0, 'Role': 'Master', 'Cpu': 0}
```

get_slot_range

Use **`get_slot_range`** to get the supported IRF member ID range.

Syntax

`get_slot_range()`

Returns

A dictionary object in the format of {'MaxSlot': *max-slot-number*, 'MinSlot': *min-slot-number* }. The *max-slot-number* argument indicates the maximum member ID. The *min-slot-number* argument indicates the minimum member ID.

Examples

# Get the supported IRF member ID range.

```
<Sysname> python
```

```
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.get_slot_range()
```

Sample output

{'MaxSlot': 2, 'MinSlot': 1}

get_standby_slot

Use **`get_standby_slot`** to get the member IDs of the subordinate devices.

Syntax

`get_standby_slot()`

Returns

A list object in one of the following formats:

- `[ ]`—The IRF fabric does not have a subordinate device.
- `[[`-1,*slot-number*`]]`—The IRF fabric has only one subordinate device.
- `[[`-1,*slot-number1*`]`,`[`-1,*slot-number2*`]`,...`]`—The IRF fabric has multiple subordinate devices.

The *slot-number* arguments indicate the member IDs of the subordinate devices.

Examples

# Get the member IDs of the subordinate devices.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.get_standby_slot()
```

Sample output

```
[[-1, 1], [-1, 2]]
```

Transfer

Use **`Transfer`** to download a file from a server.

Syntax

`Transfer`(*`protocol`*=‘’, *`host`*=‘’, *`source`*=‘’, *`dest`*=‘’, *`vrf`*=‘’,*`login_timeout`*=10, *`user`*=‘’, *`password`*=‘’)

Parameters

`protocol`: Specifies the protocol used to download a file:

- **`ftp`**—Uses FTP.
- **`tftp`**—Uses TFTP.
- **`http`**—Uses HTTP.

`host`: Specifies the IP address of the remote server.

source: Specifies the name of the file to be downloaded from the remote server.

dest: Specifies a name for the downloaded file.

vrf: Specifies the MPLS L3VPN instance to which the remote server belongs. This argument represents the VPN instance name, a case-sensitive string of 1 to 31 characters. If the server belongs to the public network, do not specify this argument.

login_timeout: Specifies the timeout for the operation, in seconds. The default is 10.

user: Specifies the username for logging in to the server.

password: Specifies the login password.

Returns

Transfer object

Examples

# Download file **test.cfg** from TFTP server 192.168.1.26.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> platformtools.Transfer('tftp', '192.168.1.26', 'test.cfg', 'flash:/test.cfg',
user='', password='')
```

Sample output

```
<platformtools.Transfer object at 0xb7f700e0>
```

send

Use **send** to generate and send a log message.

Syntax

SYSLOG.send(*digest*=‘’, *info*=‘’, *priority*=‘’)

Parameters

digest: Specifies a mnemonic for the log message. The mnemonic is a case-sensitive string of 1 to 32 characters.

info: Specifies the content of the log message, a case-sensitive string of 1 to 1024 characters.

priority: Specifies a priority for the log message, in the range of 0 to 4294967295.

Returns

None.

Usage guidelines

You must use the **SYSLOG** function to create a syslog object before you can generate and send a log message.

To display the generated log messages on the current terminal, first execute the **terminal monitor** command in user view.

For more information about the log message format, see information center configuration in *Network Management and Monitoring Configuration Guide*.

Examples

# Enable the current terminal to display log messages, and then generate and send a log message for the SNMP module. Set the mnemonic to **Test**, the content to **Try to send one message.**, and the priority to 1000.

```
<Sysname> terminal monitor
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> a = platformtools.SYSLOG('snmp')
>>> a.send('Test','Try to send one message.',1000)
```

Sample output

```
>>> %Jan  1 06:24:17:908 2019 Sysname SNMP/0/Test: Try to send one message.
```

SYSLOG

Use **SYSLOG** to create a syslog object.

Syntax

SYSLOG(*module*=‘’)

Parameters

module: Specifies the name of the feature module for which the Syslog object is created. The name is a case-insensitive string of 1 to 8 characters.

Returns

None.

Usage guidelines

You must create a syslog object for a feature module before you can generate and send a log message for that object.

Examples

# Create a syslog object named **SNMP** and assign the object to variable **a**.

```
<Sysname> python
Python 2.7.3 (default)
[GCC 4.4.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import platformtools
>>> a = platformtools.SYSLOG('snmp')
>>> a
```

Sample output

```
<platformtools.SYSLOG object at 0xb7e180e0>
```

NSFOCUS Firewall Series

NF Virtual Technologies

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for virtual technologies features, including IRF, context and Reth interface and redundancy group.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

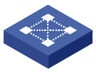
GUI conventions

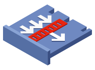
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring an IRF fabric

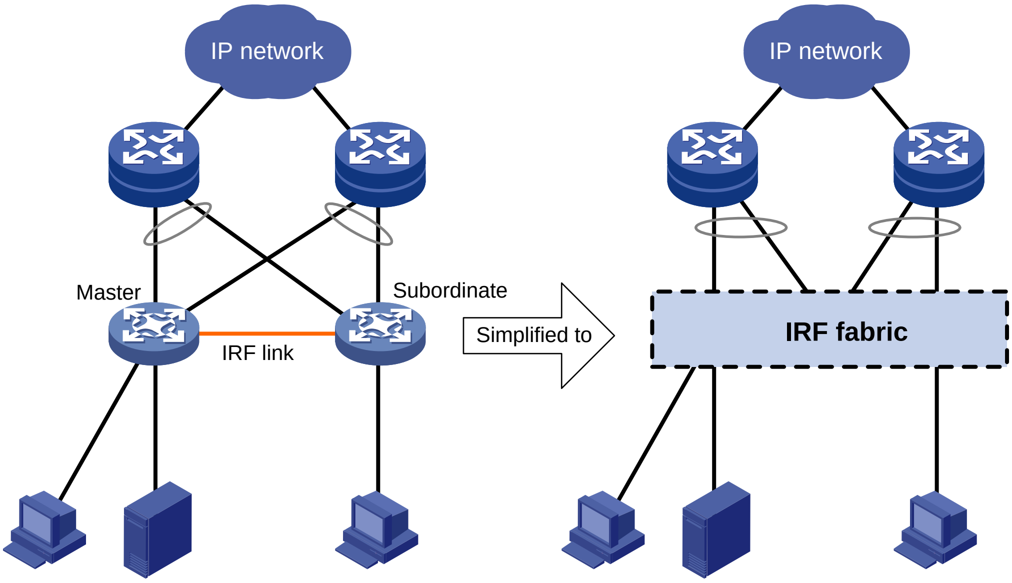
About IRF

The Intelligent Resilient Framework (IRF) technology virtualizes multiple physical devices at the same layer into one virtual fabric to provide data center class availability and scalability. IRF virtualization technology offers processing power, interaction, unified management, and uninterrupted maintenance of multiple devices.

IRF network model

Figure 1 shows an IRF fabric that has two devices, which appear as a single node to the upper-layer and lower-layer devices.

Figure 1 IRF application scenario



IRF benefits

IRF provides the following benefits:

- **Simplified topology and easy management**—An IRF fabric appears as one node and is accessible at a single IP address on the network. You can use this IP address to log in at any member device to manage all the members of the IRF fabric. In addition, you do not need to run the spanning tree feature among the IRF members.
- **1:N redundancy**—In an IRF fabric, one member acts as the master to manage and control the entire IRF fabric. All the other members process services while backing up the master. When the master fails, all the other member devices elect a new master from among them to take over without interrupting services.
- **IRF link aggregation**—You can assign several physical links between neighboring members to their IRF ports to create a load-balanced aggregate IRF connection with redundancy.
- **Multichassis link aggregation**—You can use the Ethernet link aggregation feature to aggregate the physical links between the IRF fabric and its upstream or downstream devices across the IRF members.

- **Network scalability and resiliency**—Processing capacity of an IRF fabric equals the total processing capacities of all the members. You can increase ports, network bandwidth, and processing capacity of an IRF fabric simply by adding member devices without changing the network topology.

Basic concepts

IRF member roles

IRF uses two member roles: master and standby (called subordinate throughout the documentation).

When devices form an IRF fabric, they elect a master to manage and control the IRF fabric, and all the other devices back up the master. When the master device fails, the other devices automatically elect a new master. For more information about master election, see "Master election."

IRF member ID

An IRF fabric uses member IDs to uniquely identify and manage its members. This member ID information is included as the first part of interface numbers and file paths to uniquely identify interfaces and files in an IRF fabric. Two devices cannot form an IRF fabric if they use the same member ID. A device cannot join an IRF fabric if its member ID has been used in the fabric.

Member priority

Member priority determines the possibility of a member device to be elected the master. A member with higher priority is more likely to be elected the master.

IRF port

An IRF port is a logical interface that connects IRF member devices. Every IRF-capable device has two IRF ports.

The IRF ports are named IRF-port *n*/1 and IRF-port *n*/2, where *n* is the member ID of the device. The two IRF ports are referred to as IRF-port 1 and IRF-port 2.

To use an IRF port, you must bind a minimum of one physical interface to it. The physical interfaces assigned to an IRF port automatically form an aggregate IRF link. An IRF port goes down when all its IRF physical interfaces are down.

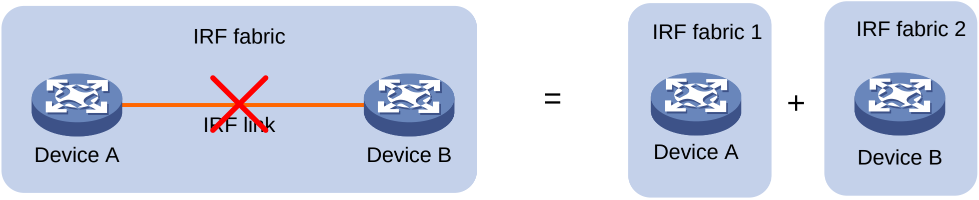
IRF physical interface

IRF physical interfaces connect IRF member devices and must be bound to an IRF port. They forward traffic between member devices, including IRF protocol packets and data packets that must travel across IRF member devices.

IRF split

IRF split occurs when an IRF fabric breaks up into multiple IRF fabrics because of IRF link failures, as shown in Figure 2. The split IRF fabrics operate with the same IP address. IRF split causes routing and forwarding problems on the network. To quickly detect a multi-active collision, configure a minimum of one MAD mechanism (see "Configuring MAD").

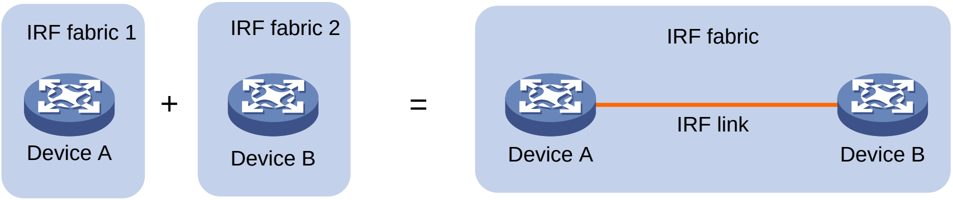
Figure 2 IRF split



IRF merge

IRF merge occurs when two split IRF fabrics reunite or when two independent IRF fabrics are united, as shown in Figure 3.

Figure 3 IRF merge



MAD

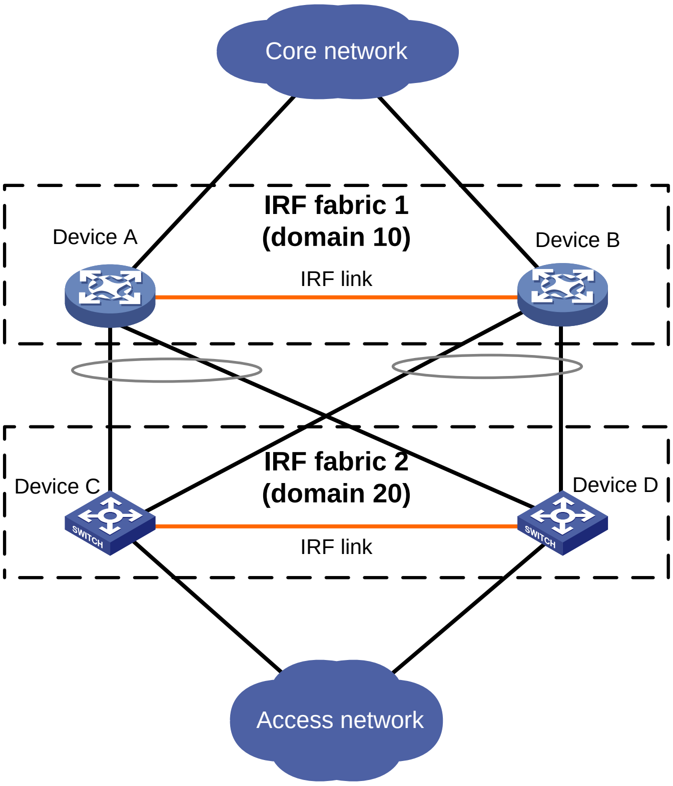
An IRF link failure causes an IRF fabric to split in two IRF fabrics operating with the same Layer 3 settings, including the same IP address. To avoid IP address collision and network problems, IRF uses multi-active detection (MAD) mechanisms to detect the presence of multiple identical IRF fabrics, handle collisions, and recover from faults.

IRF domain ID

One IRF fabric forms one IRF domain. IRF uses IRF domain IDs to uniquely identify IRF fabrics and prevent IRF fabrics from interfering with one another.

As shown in Figure 4, IRF fabric 1 contains Device A and Device B, and IRF fabric 2 contains Device C and Device D. Both fabrics use the LACP aggregate links between them for MAD. When a member device receives an extended LACPDU for MAD, it checks the domain ID to determine whether the packet is from the local IRF fabric. Then, the member device can handle the packet correctly.

Figure 4 A network that contains two IRF domains



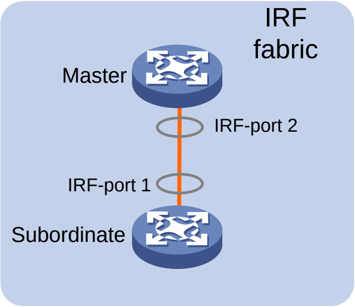
IRF network topology

An IRF fabric can use a daisy-chain topology, as shown in Figure 5.

IMPORTANT:

No relay devices are allowed between IRF member devices.

Figure 5 Daisy-chain topology



Master election

Master election occurs each time the IRF fabric topology changes in the following situations:

- The IRF fabric is established.
- The master device fails or is removed.
- The IRF fabric splits.
- Independent IRF fabrics merge.

NOTE:

Master election does not occur when split IRF fabrics merge. For information about the master device of the merged IRF fabric, see "Failure recovery."

Master election selects a master in descending order:

1. Current master, even if a new member has higher priority.

   When an IRF fabric is being formed, all members consider themselves as the master. This rule is skipped.
2. Member with higher priority.
3. Member with the longest system uptime.

   Two members are considered to start up at the same time if the difference between their startup times is equal to or less than 10 minutes. For these members, the next tiebreaker applies.
4. Member with the lowest CPU MAC address.

For the setup of a new IRF fabric, the subordinate devices must reboot to complete the setup after the master election.

For an IRF merge, devices must reboot if they are in the IRF fabric that fails the master election.

Interface naming conventions

A physical interface is numbered in the *chassis-number/slot-number/interface-index* format.

- ***chassis-number***—Member ID of the device. The default value for this argument is 1. Any change to the member ID takes effect after a reboot.
- ***slot-number***—Slot number of the interface.
- ***interface-index***—Interface index on the device. Interface index depends on the number of physical interfaces available on the device. To identify the index of a physical interface, examine its index mark on the chassis.

For example, GigabitEthernet 2/0/1 represents the first fixed physical interface on member device 2. Set its link type to trunk, as follows:

```
<Sysname> system-view
[Sysname] interface gigabitethernet 2/0/1
[Sysname-GigabitEthernet2/0/1] port link-type trunk
```

File system naming conventions

On a single-chassis fabric, you can use its storage device name to access its file system.

On a multichassis IRF fabric, you can use the storage device name to access the file system of the master. To access the file system of any other member device, use the name in the **slot***member-ID#storage-device-name* format.

For more information about storage device naming conventions, see *Fundamentals Configuration Guide*.

For example:

- To create and access the **test** folder under the root directory of the flash memory on the master switch:

  ```
  <Master> mkdir test
  Creating directory flash:/test... Done.
  <Master> cd test
  <Master> dir
  Directory of flash:/test
  The directory is empty.

  524288 KB total (29832 KB free)
  ```

- To create and access the **test** folder under the root directory of the flash memory on member device 2:

  ```
  <Master> mkdir slot2#flash:/test
  Creating directory slot2#flash:/test... Done.
  <Master> cd slot2#flash:/test
  <Master> dir
  Directory of slot2#flash:/test
  The directory is empty.

  524288 KB total (128812 KB free)
  ```

Configuration synchronization

IRF uses a strict running-configuration synchronization mechanism. In an IRF fabric, all devices obtain and run the running configuration of the master. Configuration changes are automatically propagated from the master to the remaining devices. The configuration files of these devices are retained, but the files do not take effect. The devices use their own startup configuration files only after they are removed from the IRF fabric.

As a best practice, back up the next-startup configuration file on a device before adding the device to an IRF fabric as a subordinate.

A subordinate device's next-startup configuration file might be overwritten if the master and the subordinate use the same file name for their next-startup configuration files. You can use the backup file to restore the original configuration after removing the subordinate from the IRF fabric.

For more information about configuration management, see *Fundamentals Configuration Guide*.

Multi-active handling procedure

The multi-active handling procedure includes detection, collision handling, and failure recovery.

Detection

IRF provides MAD mechanisms by extending LACP, BFD, ARP, and IPv6 ND to detect multi-active collisions. As a best practice, configure a minimum of one MAD mechanism on an IRF fabric. For more information about the MAD mechanisms and their application scenarios, see "MAD mechanisms."

For information about LACP, see Ethernet link aggregation in *Layer 2—LAN Switching Configuration Guide*. For information about BFD, see *Network Management and Monitoring Configuration Guide*. For information about ARP, see *Layer 3—IP Services Configuration Guide*. For information about ND, see IPv6 basics in *Layer 3—IP Services Configuration Guide*.

Collision handling

When MAD detects a multi-active collision, it sets all IRF fabrics except one to the Recovery state. The fabric that is not placed in Recovery state can continue to forward traffic. The Recovery-state IRF fabrics are inactive and cannot forward traffic.

LACP MAD and BFD MAD use the following process to handle a multi-active collision:

1. Compare the number of members in each fabric.
2. Set all fabrics to the Recovery state except the one that has the most members.
3. Compare the member IDs of the masters if all IRF fabrics have the same number of members.
4. Set all fabrics to the Recovery state except the one that has the lowest numbered master.
5. Shut down all common network interfaces in the Recovery-state fabrics except for the following interfaces:
   - Interfaces automatically excluded from being shut down by the system.
   - Interfaces specified by using the **`mad exclude interface`** command.

ARP MAD and ND MAD use the following process to handle a multi-active collision:

1. Compare the member IDs of the masters in the IRF fabrics.
2. Set all fabrics to the Recovery state except the one that has the lowest numbered master.
3. Take the same action on the network interfaces in Recovery-state fabrics as LACP MAD and BFD MAD.

Failure recovery

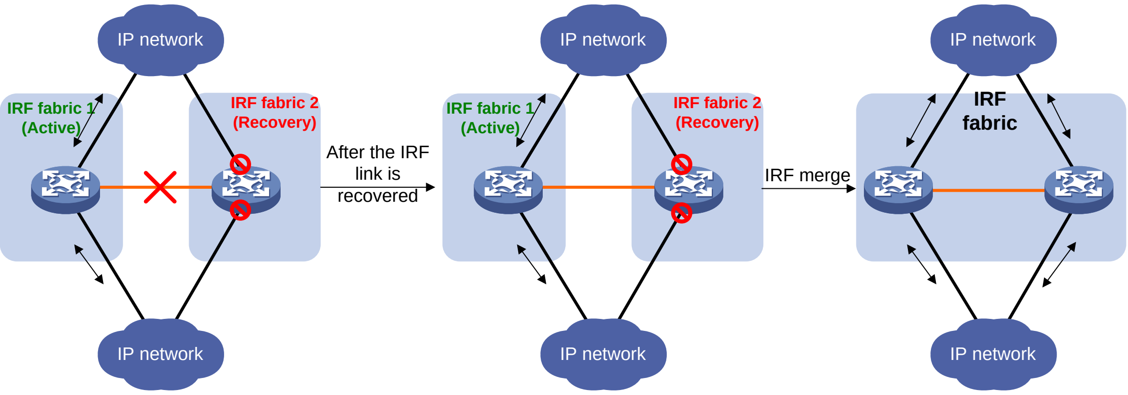
To merge two split IRF fabrics, first repair the failed IRF link and remove the IRF link failure.

After the failed IRF link between two split IRF fabrics is recovered, reboot the member devices in the inactive IRF fabric. The member devices in the inactive IRF fabric join the active IRF fabric as subordinate devices. The network interfaces that have been shut down by MAD automatically restore their original state, as shown in Figure 6.

△ **CAUTION:**

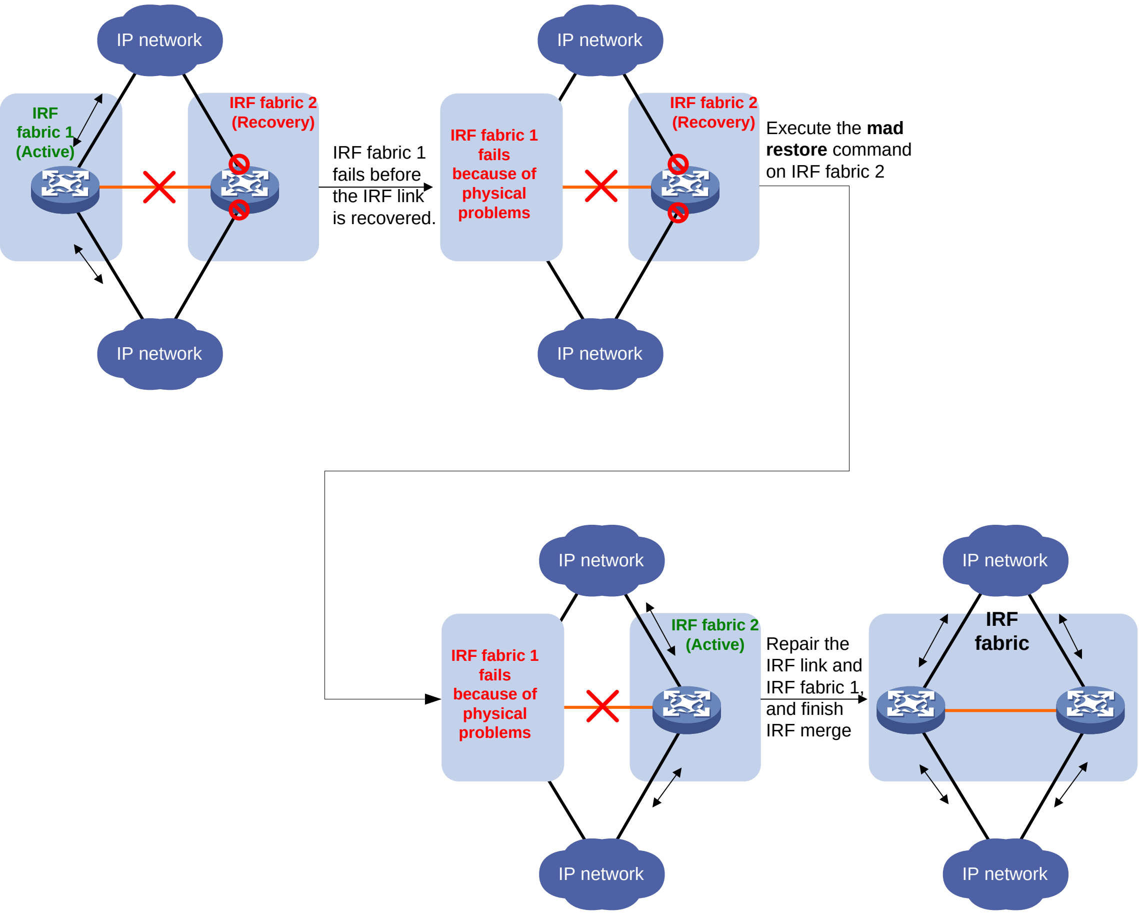
If you inadvertently reboot the active IRF fabric after the failed IRF link recovers, its member devices will join the inactive IRF fabric with their network interfaces being shut down by MAD. To restore the original states of the network interfaces in the merged IRF fabric, use the **`mad restore`** command.

Figure 6 Recovering the IRF fabric



If the active IRF fabric fails before the IRF link is recovered (see Figure 7), use the **`mad restore`** command on the inactive IRF fabric to recover the inactive IRF fabric. This command brings up all network interfaces that were shut down by MAD. After the IRF link is repaired, merge the two parts into a unified IRF fabric.

Figure 7 Active IRF fabric fails before the IRF link is recovered


IP network
IRF fabric 1 (Active)
IRF fabric 2 (Recovery)
IP network
IRF fabric 1 fails before the IRF link is recovered.
IP network
IRF fabric 1 fails because of physical problems
IRF fabric 2 (Recovery)
IP network
Execute the mad restore command on IRF fabric 2
IP network
IRF fabric 1 fails because of physical problems
IRF fabric 2 (Active)
IP network
Repair the IRF link and IRF fabric 1, and finish IRF merge
IP network
IRF fabric
IP network

MAD mechanisms

IRF provides MAD mechanisms by extending LACP, BFD, ARP, and IPv6 ND.

Table 1 compares the MAD mechanisms and their application scenarios.

Table 1 Comparison of MAD mechanisms

MAD mechanism	Advantages	Disadvantages	Application scenarios
LACP MAD	• Detection speed is fast. • Runs on existing aggregate links without requiring MAD-dedicated physical links or Layer 3 interfaces.	Requires an intermediate device that supports extended LACP for MAD.	Link aggregation is used between the IRF fabric and its upstream or downstream device.
BFD MAD	• Detection speed is fast. • Intermediate device, if used, can come from any vendor.	Requires MAD dedicated physical links and Layer 3 interfaces, which cannot be used for transmitting user traffic.	• No special requirements for network scenarios. • If no intermediate device is used, this mechanism is only suitable for IRF fabrics that have only two members that are geographically close to one another.
ARP MAD	• No intermediate device is required. • Intermediate device, if used, can come from any vendor. • Does not require MAD dedicated ports.	• Detection speed is slower than BFD MAD and LACP MAD. • The spanning tree feature must be enabled.	Spanning tree-enabled non-link aggregation IPv4 network scenarios.
ND MAD	• No intermediate device is required. • Intermediate device, if used, can come from any vendor. • Does not require MAD dedicated ports.	• Detection speed is slower than BFD MAD and LACP MAD. • The spanning tree feature must be enabled.	Spanning tree-enabled non-link aggregation IPv6 network scenarios.

LACP MAD

As shown in Figure 8, LACP MAD has the following requirements:

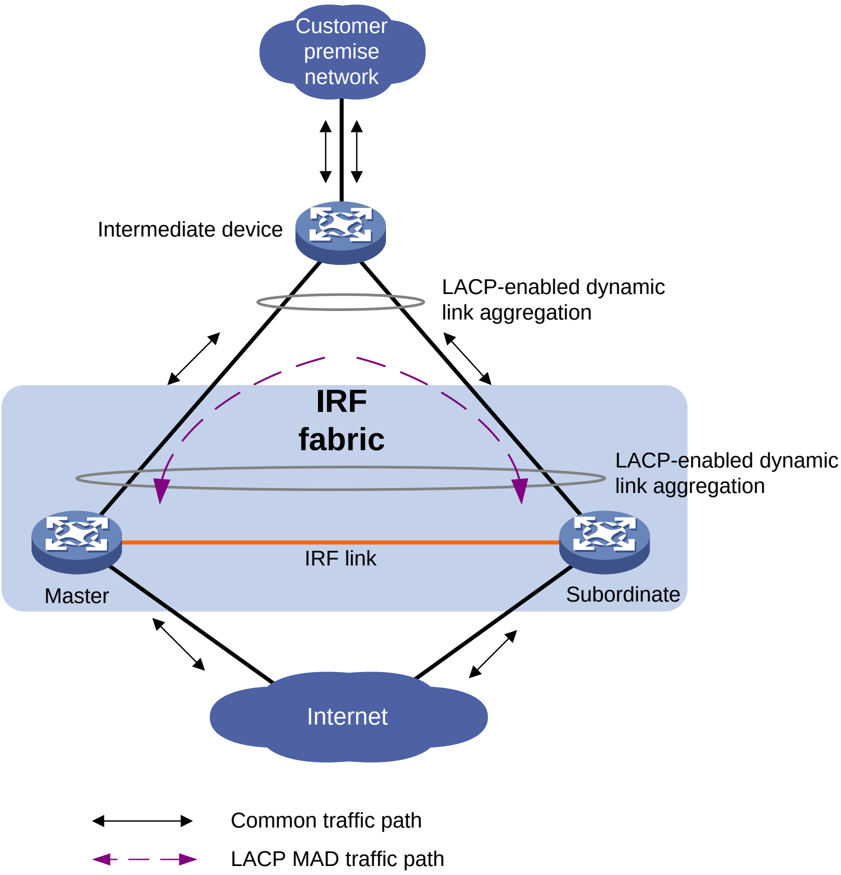
- Every IRF member must have a link with an intermediate device.
- All the links form a dynamic link aggregation group.
- The intermediate device must be a device that supports extended LACP for MAD.

The IRF member devices send extended LACPDUs that convey a domain ID and an active ID (the member ID of the master). The intermediate device transparently forwards the extended LACPDUs received from one member device to all the other member devices.

- If the domain IDs and active IDs sent by all the member devices are the same, the IRF fabric is integrated.

- If the extended LACPDUs convey the same domain ID but different active IDs, a split has occurred. LACP MAD handles this situation as described in "Collision handling."

Figure 8 LACP MAD scenario



BFD MAD

BFD MAD detects multi-active collisions by using BFD.

You can use common Ethernet ports for BFD MAD.

If management Ethernet ports are used, BFD MAD has the following requirements:

- An intermediate device is required and each IRF member device must have a BFD MAD link to the intermediate device.
- Each member device is assigned a MAD IP address on the master's management Ethernet port.

If common Ethernet ports are used, BFD MAD has the following requirements:

- If an intermediate device is used, each member device must have a BFD MAD link to the intermediate device. If no intermediate device is used, all member devices must have a BFD MAD link to each other. As a best practice, use an intermediate device to connect IRF member devices if the IRF fabric has more than two member devices. A full mesh of IRF members might cause broadcast loops.
- Ports on BFD MAD links are assigned to a VLAN (or Layer 3 aggregate interface) used for BFD MAD. Each member device is assigned a MAD IP address on the VLAN interface (or Layer 3 aggregate interface).

As a best practice, preferentially use management Ethernet ports for BFD MAD.

The BFD MAD links and BFD MAD VLAN (or Layer 3 aggregate interface) must be dedicated. Do not use BFD MAD links or BFD MAD VLAN (or Layer 3 aggregate interface) for any other purposes.

When you use a Layer 3 aggregate interface for BFD MAD, make sure its member ports do not exceed the maximum number of Selected ports allowed for an aggregation group. If the number of member ports exceeds the maximum number of Selected ports, some member ports cannot become Selected. BFD MAD will be unable to work correctly and its state will change to Faulty. For more information about setting the maximum number of Selected ports for an aggregation group, see Ethernet link aggregation in *Layer 2—LAN Switching Configuration Guide*.

NOTE:

- The MAD addresses identify the member devices and must belong to the same subnet.
- Of all management Ethernet ports on an IRF fabric, only the master's management Ethernet port is accessible.

Figure 9 shows a typical BFD MAD scenario that uses an intermediate device. On the intermediate device, assign the ports on the BFD MAD links to the same VLAN.

Figure 10 shows a typical BFD MAD scenario that does not use an intermediate device.

With BFD MAD, the master attempts to establish BFD sessions with other member devices by using its MAD IP address as the source IP address.

- If the IRF fabric is integrated, only the MAD IP address of the master takes effect. The master cannot establish a BFD session with any other member. If you execute the **`display bfd session`** command, the state of the BFD sessions is **Down**.
- When the IRF fabric splits, the IP addresses of the masters in the split IRF fabrics take effect. The masters can establish a BFD session. If you execute the **`display bfd session`** command, the state of the BFD session between the two devices is **Up**.

Figure 9 BFD MAD scenario with an intermediate device

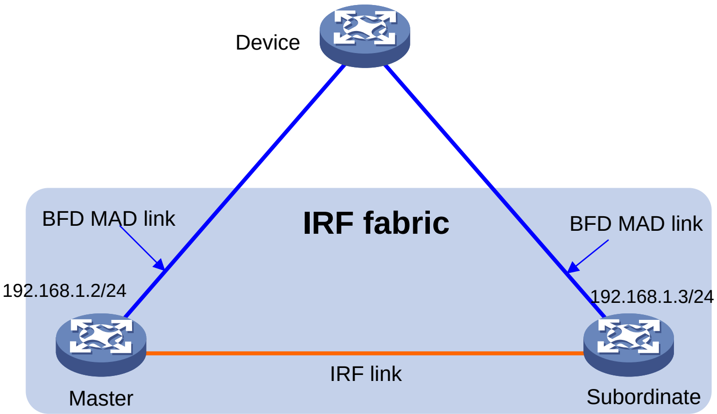
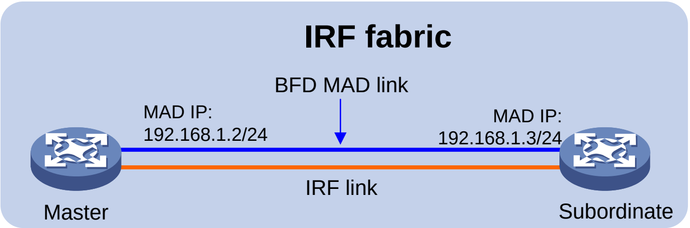


Figure 10 BFD MAD scenario without an intermediate device



ARP MAD

ARP MAD detects multi-active collisions by using extended ARP packets that convey the IRF domain ID and the active ID.

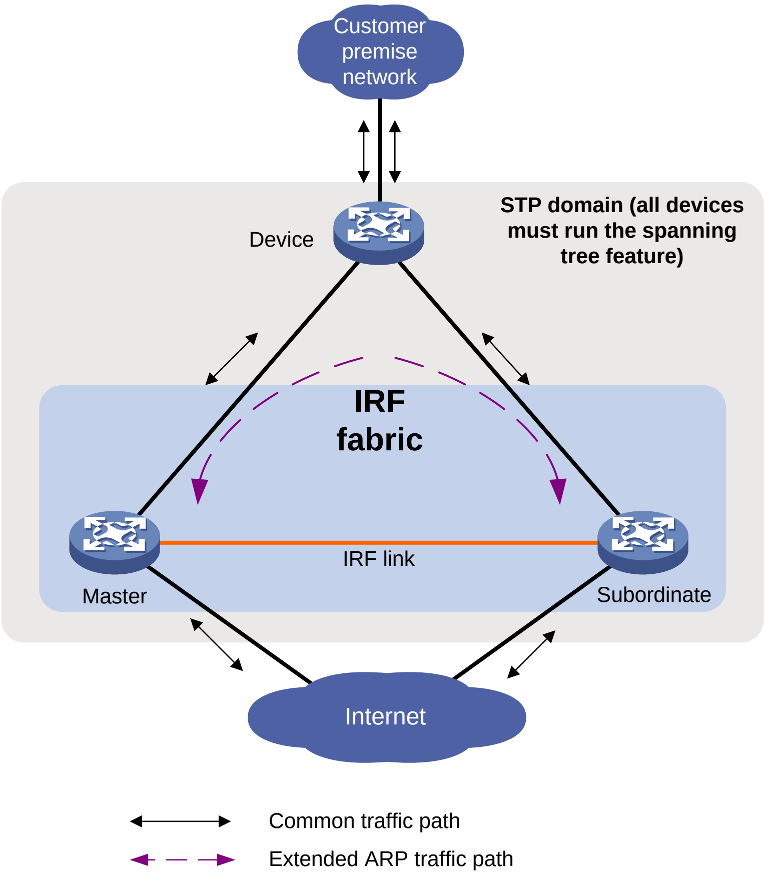
ARP MAD can work with or without an intermediate device. Make sure the following requirements are met:

- If an intermediate device is used, connect each IRF member device to the intermediate device, as shown in Figure 11. Run the spanning tree feature between the IRF fabric and the intermediate device. In this situation, data links can be used.
- If no intermediate device is used, connect each IRF member device to all other member devices. In this situation, IRF links cannot be used for ARP MAD.

Each IRF member compares the domain ID and the active ID (the member ID of the master) in incoming extended ARP packets with its domain ID and active ID.

- If the domain IDs are different, the extended ARP packet is from a different IRF fabric. The device does not continue to process the packet with the MAD mechanism.
- If the domain IDs are the same, the device compares the active IDs.
  - If the active IDs are different, the IRF fabric has split.
  - If the active IDs are the same, the IRF fabric is integrated.

Figure 11 ARP MAD scenario



ND MAD

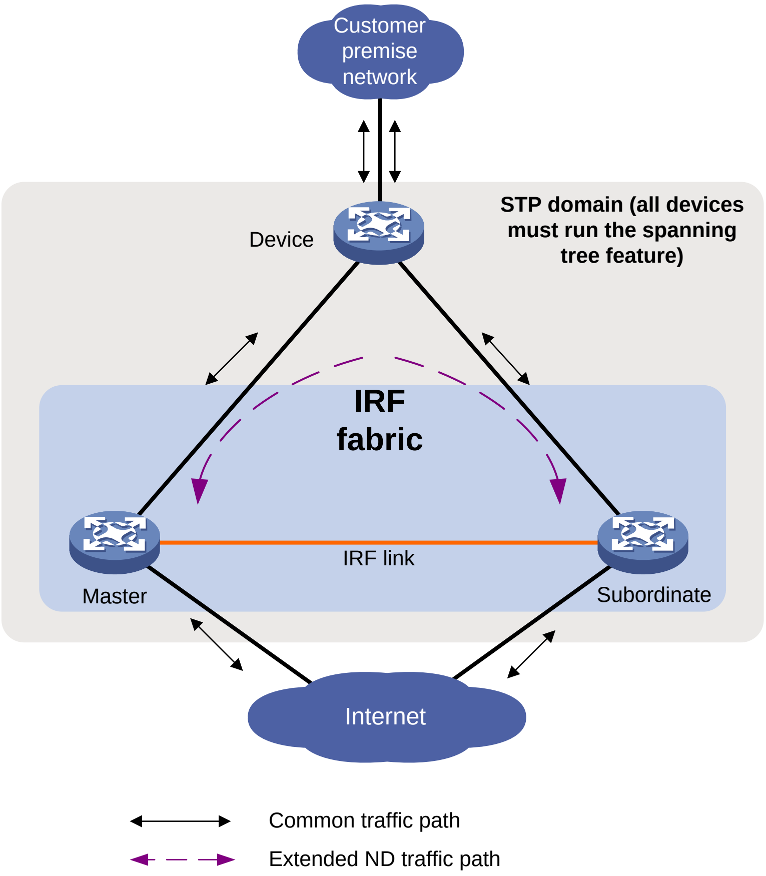
ND MAD detects multi-active collisions by using NS packets to transmit the IRF domain ID and the active ID.

You can set up ND MAD links between neighbor IRF member devices or between each IRF member device and an intermediate device (see Figure 12). If an intermediate device is used, you must also run the spanning tree protocol between the IRF fabric and the intermediate device.

Each IRF member device compares the domain ID and the active ID (the member ID of the master) in incoming NS packets with its domain ID and active ID.

- If the domain IDs are different, the NS packet is from a different IRF fabric. The device does not continue to process the packet with the MAD mechanism.
- If the domain IDs are the same, the device compares the active IDs.
  - If the active IDs are different, the IRF fabric has split.
  - If the active IDs are the same, the IRF fabric is integrated.

Figure 12 ND MAD scenario



Restrictions: Hardware compatibility with IRF

Models	IRF compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: IRF configuration

Support for Web-based configuration

You can configure only basic IRF settings from the Web interface. To configure IRF parameters not available on the Web interface, access the CLI.

Hardware compatibility with IRF

The device can form an IRF fabric only with devices of the same model.

Software requirements for IRF

All IRF member devices must run the same software image version. Make sure the software auto-update feature is enabled on all member devices.

IRF fabric size

An IRF fabric can contain a maximum of two member devices.

Candidate IRF physical interfaces

As a best practice, use high-speed ports as IRF physical interfaces.

Do not use the following ports as IRF physical interfaces:

- Console ports.
- Management ports. The interface type identifier for management ports is M-GigabitEthernet. The port identifier for a management port has a suffix of "/MGMT" on the panel.
- A port that has been assigned by default or manually to a bridge instance enabled with security service bypass..

As a best practice, do not bind a Ten-GigabitEthernet (XGE) fiber port to an IRF port if a gigabit fiber transceiver module is installed in it. The port identifier for a Ten-GigabitEthernet fiber port is **10GBASE-R** on the panel.

To use a port as an IRF physical interface, use the **`port group interface`** command to bind the port to an IRF port.

Use Table 2 to identify which ports on your device or module can act as IRF physical interfaces.

Table 2 NFNX series and candidate IRF physical interfaces matrix

Models	Candidate IRF physical interfaces
NFNX5-HD6480	• XGE 1/0/14 to XGE 1/0/17. • XGE 1/0/20. • XGE 1/0/21.
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280	• GE 1/0/0 to GE 1/0/23. • XGE 1/0/24 to XGE 1/0/29.
NFNX3-HDB1780, NFNX3-HDB3080	• GE 1/0/0 to GE 1/0/25. • XGE 1/0/26. • XGE 1/0/27.
NFNX3-HDB1180, NFNX3-HDB1480	• GE 1/0/4 to GE 1/0/21. • XGE 1/0/30. • XGE 1/0/31.
NFNX3-HDB680, NFNX3-HDB1080	IRF not supported.

Transceiver modules and cables selection for IRF

When you select transceiver modules and cables, follow these restrictions and guidelines:

- Use Category 5 (or above) twisted-pair cables to connect 10/100/1000Mbps Ethernet ports for a short-distance connection.
- Use SFP or SFP+ DAC cables to connect SFP or SFP+ ports for a short-distance connection.
- Use SFP or SFP+ transceiver modules and fibers to connect SFP or SFP+ ports for a long-distance connection.
- The transceiver modules at the two ends of an IRF link must be the same type.

For more information about the transceiver modules and DAC cables, see the device installation guide and *NSFOCUS Transceiver Modules User Guide*.

NOTE:

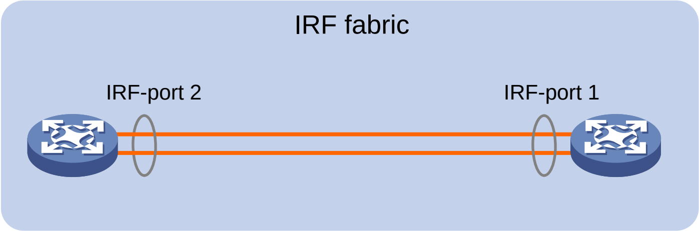
The transceiver modules and DAC cables available for the device are subject to change over time. For the most up-to-date list of transceiver modules and DAC cables, contact your NSFOCUS sales representative.

IRF port connection

When you connect two neighboring IRF members, follow these restrictions and guidelines:

- You must connect the physical interfaces of IRF-port 1 on one member to the physical interfaces of IRF-port 2 on the other.
- For high availability, bind multiple physical interfaces to an IRF port.
- No relay devices are allowed between neighboring members.

Figure 13 Connecting IRF physical interfaces



IRF physical interface configuration restrictions and guidelines

Command configuration restrictions

On a physical interface bound to an IRF port, you can execute only the following commands:

- Interface commands, including:
  - **`description`**.
  - **`flow-interval`**.
  - **`shutdown`**.

  For more information about these commands, see Ethernet interface configuration in *Interface Command Reference*.
- LLDP commands, including:

- ○ **lldp admin-status**.
- ○ **lldp check-change-interval**.
- ○ **lldp enable**.
- ○ **lldp encapsulation snap**.
- ○ **lldp notification remote-change enable**.
- ○ **lldp tlv-enable**.

For more information about these commands, see *Layer 2—LAN Switching Command Reference*.

- The **mirroring-group reflector-port** command, which specifies the physical interface as a reflector port for remote mirroring. For more information about this command, see port mirroring in *Network Management and Monitoring Command Reference*.

IMPORTANT:

Do not execute the **mirroring-group reflector-port** command on an IRF physical interface if that interface is the only member interface of an IRF port. Doing so will split the IRF fabric, because this command also removes the binding of the physical interface and IRF port.

Feature compatibility and configuration restrictions with IRF

ACL

To form an IRF fabric, all member devices in the IRF fabric must have the same settings for the ACL hardware mode. For more information about hardware-based ACLs, see *ACL and QoS Configuration Guide*.

Licensing requirements for IRF

For a license-based feature to run correctly on an IRF fabric, make sure the licenses installed for the feature on all member devices are the same. For more information about feature licensing, see *Fundamentals Configuration Guide*.

Configuration rollback restrictions

The configuration rollback feature cannot roll back the following IRF settings:

- Member device description (set by using the **irf member description** command).
- Member device priority (set by using the **irf member priority** command).
- IRF physical interface and IRF port bindings (set by using the **port group interface** command).

For more information about the configuration rollback feature, see configuration file management in *Fundamentals Configuration Guide*.

IRF tasks at a glance

To configure IRF, perform the following tasks:

1. Setting up an IRF fabric
2. Configuring MAD

   Configure a minimum of one MAD mechanism on an IRF fabric.

   - Configuring LACP MAD
   - Configuring BFD MAD
   - Configuring ARP MAD
   - Configuring ND MAD
   - Excluding interfaces from the shutdown action upon detection of multi-active collision

     This feature excludes an interface from the shutdown action for management or other special purposes when an IRF fabric transits to the Recovery state.
   - Recovering an IRF fabric
3. (Optional.) Optimizing IRF settings for an IRF fabric
   - Configuring a member device description
   - Configuring IRF link load sharing mode
   - Configuring the IRF bridge MAC address
   - Enabling software auto-update for software image synchronization

     This feature automatically synchronizes the current software images of the master to devices that are attempting to join the IRF fabric.
   - Removing an expansion interface card that has IRF physical interfaces
   - Replacing an expansion interface card that has IRF physical interfaces

Planning the IRF fabric setup

Consider the following items when you plan an IRF fabric:

- Hardware compatibility and restrictions.
- IRF fabric size.
- Master device.
- Member ID and priority assignment scheme.
- Fabric topology and cabling scheme.
- IRF physical interfaces.

Setting up an IRF fabric

IRF setup tasks at a glance

To set up an IRF fabric, perform the following tasks:

1. Configure member IDs, priorities, and IRF physical interfaces separately.
   a. Assigning a member ID to each IRF member device
   b. (Optional.) Specifying a priority for each member device
   c. Binding physical interfaces to IRF ports

   Skip these tasks if you configure member IDs, priorities, domain ID, and IRF physical interfaces in bulk.
2. Bulk-configuring basic IRF settings for a member device

   Skip this task if you configure member IDs, priorities, domain ID, and IRF physical interfaces separately.
3. Connecting IRF physical interfaces
4. Accessing the IRF fabric

Assigning a member ID to each IRF member device

Restrictions and guidelines

To create an IRF fabric, you must assign a unique IRF member ID to each member device.

The new member ID of a device takes effect at a reboot. After the device reboots, the settings on all member ID-related physical resources (including common physical network interfaces) are removed, regardless of whether you have saved the configuration.

Procedure

1. Enter system view.

   `system-view`

2. Assign a member ID to a member device.

   `irf member` *`member-id`* **`renumber`** *`new-member-id`*

   The default IRF member ID is 1.

   △ **CAUTION:**

   An IRF member ID change can invalidate member ID-related settings and cause data loss. Make sure you fully understand its impact on the live network.

3. (Optional.) Save the configuration.

   `save`

   If you have bound physical interfaces to IRF ports or assigned member priority, you must perform this step for these settings to take effect after the reboot.

4. Return to user view.

   `quit`

5. Reboot the device.

   `reboot` `[` **`slot`** *`slot-number`* `] [` **`force`** `]`

Specifying a priority for each member device

About this task

IRF member priority represents the possibility for a device to be elected the master in an IRF fabric. A larger priority value indicates a higher priority.

A change to member priority affects the election result at the next master election, but it does not cause an immediate master re-election.

Procedure

1. Enter system view.

   `system-view`

2. Specify a priority for the device.

   `irf member` *`member-id`* **`priority`** *`priority`*

   The default IRF member priority is 1.

Binding physical interfaces to IRF ports

Restrictions and guidelines

Select qualified physical interfaces as IRF physical interfaces as described in "Candidate IRF physical interfaces."

Make sure the IRF physical interfaces of an IRF port use the same binding mode.

After binding physical interfaces to IRF ports for the first time, you must use the **irf-port-configuration active** command to activate the settings on the IRF ports.

The system activates the IRF port settings automatically only in the following situations:

- The configuration file that the device starts with contains IRF port bindings.
- You are adding physical interfaces to an IRF port (in UP state) after an IRF fabric is formed.

Procedure

1. Enter system view.

   system-view

2. Enter the interface view or interface range view of an IRF physical interface or a range of IRF physical interfaces, respectively.

   - Enter Layer 2 or Layer 3 Ethernet interface view.

     interface *interface-type interface-number*

   - Enter interface range view. Choose one of the following commands:

     interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

     interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

   To shut down a range of IRF physical interfaces, enter interface range view.

   To shut down one IRF physical interface, enter its interface view.

3. Shut down the physical interfaces.

   shutdown

   By default, a physical interface is not administratively down.

   You must always shut down a physical interface before binding it to an IRF port or removing the binding. If the system prevents you from shutting down an interface, follow the system instructions to disable its peer interface.

4. Return to system view.

   quit

5. Enter IRF port view.

   irf-port *member-id/irf-port-number*

6. Bind each physical interface to the IRF port.

   port group interface *interface-type interface-number*

   By default, no physical interfaces are bound to an IRF port.

   Repeat this step to assign multiple physical interfaces to the IRF port.

7. Return to system view.

   quit

8. Enter the interface view or interface range view of an IRF physical interface or a range of IRF physical interfaces, respectively.

   - Enter Layer 2 or Layer 3 Ethernet interface view.

interface *interface-type interface-number*

  - Enter interface range view. Choose one of the following commands:

    interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

    interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

9. Bring up the physical interfaces.

   undo shutdown

10. Return to system view.

    quit

11. Save the configuration.

    save

    Activating IRF port configurations causes IRF merge and reboot. To avoid data loss, save the running configuration to the startup configuration file before you perform the operation.

12. Activate the IRF port settings.

    irf-port-configuration active

Bulk-configuring basic IRF settings for a member device

About this task

Use the easy IRF feature to bulk-configure basic IRF settings for a member device, including the member ID, domain ID, priority, and IRF port bindings.

The easy IRF feature provides the following configuration methods:

- **Interactive method**—Enter the **easy-irf** command without parameters. The system will guide you to set the parameters step by step.
- **Non-interactive method**—Enter the **easy-irf** command with parameters.

As a best practice, use the interactive method if you are new to IRF.

Restrictions and guidelines

△ **CAUTION:**

- Use caution when you change the member ID of an IRF member device. An IRF member ID uniquely identifies a device in an IRF fabric . An IRF member ID change can invalidate member ID-related settings, including interface and file path settings, and cause data loss. Make sure you fully understand its impact on the live network.
- The member device reboots immediately after you specify a new member ID for it. Make sure you are aware of the impact of this operation on the network.

If you execute the **easy-irf** command multiple times, the following settings take effect:

- The most recent settings for the member ID, domain ID, and priority.
- IRF port bindings added through repeated executions of the command. To remove an IRF physical interface from an IRF port, you must use the **undo port group interface** command in IRF port view.

If you specify IRF physical interfaces by using the interactive method, you must also follow these restrictions and guidelines:

- Do not enter spaces between the interface type and interface number.
- Use a comma (,) to separate two physical interfaces. No spaces are allowed between interfaces.

Procedure

1. Enter system view.

   system-view

2. Bulk-configure basic IRF settings for the device.

   easy-irf [**member** *member-id* [**renumber** *new-member-id*] **domain** *domain-id* [**priority** *priority*] [**irf-port1** *interface-list1*] [**irf-port2** *interface-list2*]]

   Make sure the new member ID is unique in the IRF fabric to which the device will be added.

Connecting IRF physical interfaces

Follow the restrictions in "IRF port connection" to connect IRF physical interfaces as well as based on the topology and cabling scheme. The devices perform master election. The member devices that fail the master election automatically reboot to form an IRF fabric with the master device.

Accessing the IRF fabric

The IRF fabric appears as one device after it is formed. You configure and manage all IRF members at the CLI of the master. All settings you have made are automatically propagated to the IRF members.

The following methods are available for accessing an IRF fabric:

- **Local login**—Log in through the AUX or console port of any member device.
- **Remote login**—Log in at a Layer 3 interface on any member device by using methods including Telnet and SNMP.

When you log in to an IRF fabric, you are placed at the CLI of the master, regardless of at which member device you are logged in.

For more information, see login configuration in *Fundamentals Configuration Guide*.

Configuring MAD

Restrictions and guidelines for MAD configuration

VLAN interface compatibility

Do not configure MAD on VLAN interfaces.

MAD mechanism compatibility

As a best practice, configure a minimum of one MAD mechanism on an IRF fabric for prompt IRF split detection. Because MAD mechanisms use different collision handling processes, follow these restrictions and guidelines when you configure multiple MAD mechanisms on an IRF fabric:

- Do not configure LACP MAD together with ARP MAD or ND MAD.
- Do not configure BFD MAD together with ARP MAD or ND MAD.

Assigning IRF domain IDs

An IRF fabric has only one IRF domain ID.

You can change the IRF domain ID by using the following commands: **irf domain**, **mad enable**, **mad arp enable**, or **mad nd enable**. The IRF domain IDs configured by using these commands overwrite each other.

If LACP MAD, ARP MAD, or ND MAD runs between two IRF fabrics, assign each fabric a unique IRF domain ID. (For BFD MAD, this task is optional.)

Actions on interfaces shut down by MAD

To prevent a multi-active collision from causing network issues, avoid using the **undo shutdown** command to bring up the interfaces shut down by a MAD mechanism on a Recovery-state IRF fabric.

Configuring LACP MAD

1. Enter system view.

   system-view

2. Assign a domain ID to the IRF fabric.

   irf domain *domain-id*

   The default IRF domain ID is 0.

   ⚠ **CAUTION:**

   Changing the IRF domain ID of an IRF member device will remove that member device from the IRF fabric. This member device will be unable to exchange IRF protocol packets with the remaining member devices in the IRF fabric.

3. Create an aggregate interface and enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.

     interface route-aggregation *interface-number*

   Perform this step also on the intermediate device.

4. Configure the aggregation group to operate in dynamic aggregation mode.

   link-aggregation mode dynamic

   By default, an aggregation group operates in static aggregation mode.

   LACP MAD takes effect only on dynamic aggregate interfaces.

   Perform this step also on the intermediate device.

5. Enable LACP MAD.

   mad enable

   By default, LACP MAD is disabled.

6. Return to system view.

   quit

7. Enter Ethernet interface view or interface range view.
   - Enter Ethernet interface view.

     interface *interface-type interface-number*
   - Enter interface range view. Choose one of the following commands:

     interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

     interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

   To assign a range of ports to the aggregation group, enter interface range view.

   To assign one port to the aggregation group, enter Ethernet interface view.

8. Assign the Ethernet port or the range of Ethernet ports to the specified aggregation group.

   port link-aggregation group *group-id*

   Multichassis link aggregation is allowed.

   Perform this step also on the intermediate device.

Configuring BFD MAD

Restrictions and guidelines for configuring BFD MAD

As a best practice, use the following procedure to set up BFD MAD:

1. Choose a BFD MAD link scheme as described in "BFD MAD."
2. Configure BFD MAD.
3. Connect the BFD MAD links.

When you configure BFD MAD on a Layer 3 aggregate interface, follow these restrictions and guidelines:

Category	Restrictions and guidelines
BFD MAD-enabled Layer 3 aggregate interface	• Make sure the Layer 3 aggregate interface operates in static aggregation mode. • Make sure the member ports in the aggregation group do not exceed the maximum number of Selected ports allowed for an aggregation group. If the number of member ports exceeds the maximum number of Selected ports, some member ports cannot become Selected. BFD MAD will be unable to work correctly and its state will change to Faulty.
BFD MAD VLAN	If a switch is used as the intermediate device, use a VLAN as the BFD MAD VLAN on the intermediate device. • If the intermediate device acts as a BFD MAD intermediate device for multiple IRF fabrics, assign different BFD MAD VLANs to the IRF fabrics. • Assign the ports on the BFD MAD links for the same IRF fabric to the same VLAN. Do not assign the ports to an aggregate interface. If the ports are hybrid ports, make sure these ports are untagged members of their PVIDs. • Do not use the BFD MAD VLAN on the intermediate device for any purposes other than BFD MAD. • Make sure the BFD MAD VLAN on the intermediate device contains only ports on the BFD MAD links. Exclude a port from the BFD MAD VLAN if that port is not on a BFD MAD link. If you have assigned that port to all VLANs by using the **port trunk permit vlan all** command, use the **undo port trunk permit** command to exclude that port from the BFD MAD VLAN.
BFD MAD-enabled Layer 3 aggregate interface and feature compatibility	Use only the **mad bfd enable** and **mad ip address** commands on the BFD MAD-enabled interface. If you configure other features, both BFD MAD and other features on the interface might run incorrectly.
MAD IP address	• To avoid network issues, only use the **mad ip address** command to configure IP addresses on the BFD MAD-enabled interface. Do not configure an IP address by using the **ip address** command on the BFD MAD-enabled interface. • Make sure all the MAD IP addresses are on the same subnet.

Configuring BFD MAD on a Layer 3 aggregate interface

1. Enter system view.

   system-view

2. (Optional.) Assign a domain ID to the IRF fabric.

   irf domain *domain-id*

   By default, the domain ID of an IRF fabric is 0.

> **CAUTION:**
>
> Changing the IRF domain ID of an IRF member device will remove that member device from the IRF fabric. This member device will be unable to exchange IRF protocol packets with the remaining member devices in the IRF fabric.

3. Create a Layer 3 aggregate interface for BFD MAD.

   interface route-aggregation *interface-number*

4. Return to system view.

   quit

5. Enter interface view or interface range view.
   - Enter Ethernet interface view.

     interface *interface-type interface-number*
   - Enter interface range view. Choose one of the following commands:

     interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

     interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

   To assign a range of ports to the aggregation group for the aggregate interface, enter interface range view.

   To assign one port to the aggregation group for the aggregate interface, enter Ethernet interface view.

6. Assign the port or the range of ports to the aggregation group for the aggregate interface.

   port link-aggregation group *number*

7. Return to system view.

   quit

8. Enter Layer 3 aggregate interface view.

   interface route-aggregation *interface-number*

9. Enable BFD MAD.

   mad bfd enable

   By default, BFD MAD is disabled.

10. Assign a MAD IP address to a member device on the Layer 3 aggregate interface.

    mad ip address *ip-address* { *mask* | *mask-length* } **member** *member-id*

    By default, no MAD IP addresses are configured on aggregate interfaces.

    Repeat this step to assign a MAD IP address to each member device on the aggregate interface.

Configuring ARP MAD

Restrictions and guidelines

As a best practice, use the following procedure to set up ARP MAD:

1. Choose an ARP MAD link scheme as described in "ARP MAD."
2. Configure ARP MAD.

3. Connect the ARP MAD links if you are not using existing data links as ARP MAD links.

When you configure ARP MAD, follow these restrictions and guidelines:

Category	Restrictions and guidelines
ARP MAD VLAN	• Do not enable ARP MAD on VLAN-interface 1. • If you are using an intermediate device, perform the following tasks: ○ On the IRF fabric and the intermediate device, create a VLAN for ARP MAD. ○ On the IRF fabric and the intermediate device, assign the ports of ARP MAD links to the ARP MAD VLAN. ○ On the IRF fabric, create a VLAN interface for the ARP MAD VLAN. • Do not use the ARP MAD VLAN for any other purposes.
ARP MAD and feature configuration	If an intermediate device is used, make sure the following requirements are met: • Run the spanning tree feature between the IRF fabric and the intermediate device to ensure that there is only one ARP MAD link in forwarding state. For more information about the spanning tree feature and its configuration, see *Layer 2—LAN Switching Configuration Guide*. • Enable the IRF fabric to change its bridge MAC address as soon as the address owner leaves. • If the intermediate device is also an IRF fabric, assign the two IRF fabrics different domain IDs for correct split detection.

Procedure

1. Enter system view.

   `system-view`

2. Assign a domain ID to the IRF fabric.

   `irf domain` *`domain-id`*

   The default IRF domain ID is 0.

CAUTION:

Changing the IRF domain ID of an IRF member device will remove that member device from the IRF fabric. This member device will be unable to exchange IRF protocol packets with the remaining member devices in the IRF fabric.

3. Configure the IRF bridge MAC address to change as soon as the address owner leaves.

   `undo irf mac-address persistent`

   By default, the IRF bridge MAC address remains unchanged for 6 minutes after the address owner leaves.

4. Create a VLAN dedicated to ARP MAD.

   `vlan` *`vlan-id`*

   By default, only VLAN 1 exists.

5. Return to system view.

   `quit`

6. Enter Ethernet interface view or interface range view.
   - Enter Ethernet interface view.

     `interface` *`interface-type interface-number`*

   - Enter interface range view. Choose one of the following commands:

interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

To assign a range of ports to the ARP MAD VLAN, enter interface range view.

To assign one port to the ARP MAD VLAN, enter Ethernet interface view.

7. Assign the port or the range of ports to the ARP MAD VLAN.
   - Assign the ports to the VLAN as access ports.

     port access vlan *vlan-id*
   - Assign the ports to the VLAN as trunk ports.

     port trunk permit vlan *vlan-id*
   - Assign the ports to the VLAN as hybrid ports.

     port hybrid vlan *vlan-id* { **tagged** | **untagged** }

   The link type of ARP MAD ports can be access, trunk, or hybrid.

   The default link type of a port is access.
8. Return to system view.

   quit
9. Enter VLAN interface view.

   interface vlan-interface *vlan-interface-id*
10. Assign the interface an IP address.

    ip address *ip-address* { *mask* | *mask-length* }

    By default, no IP addresses are assigned to any VLAN interfaces.
11. Enable ARP MAD.

    mad arp enable

    By default, ARP MAD is disabled.

Configuring ND MAD

Restrictions and guidelines

When you use ND MAD, follow these guidelines:

- If an intermediate device is used, you can use common data links as ND MAD links. If no intermediate device is used, set up dedicated ND MAD links between IRF member devices.
- If an intermediate device is used, make sure the following requirements are met:
  - Run the spanning tree feature between the IRF fabric and the intermediate device. Make sure there is only one ND MAD link in forwarding state. For more information about the spanning tree feature and its configuration, see *Layer 2—LAN Switching Configuration Guide*.
  - Enable the IRF fabric to change its bridge MAC address as soon as the address owner leaves.
  - If the intermediate device is also an IRF fabric, assign the two IRF fabrics different domain IDs for correct split detection.

Procedure

1. Enter system view.

   system-view
2. Assign a domain ID to the IRF fabric.

irf domain *domain-id*

The default IRF domain ID is 0.

CAUTION:

Changing the IRF domain ID of an IRF member device will remove that member device from the IRF fabric. This member device will be unable to exchange IRF protocol packets with the remaining member devices in the IRF fabric.

3. Configure the IRF bridge MAC address to change as soon as the address owner leaves.

   undo irf mac-address persistent

   By default, the IRF bridge MAC address remains unchanged for 6 minutes after the address owner leaves.
4. Create a VLAN dedicated to ND MAD.

   vlan *vlan-id*

   By default, only VLAN 1 exists.

   Do not configure ND MAD on VLAN-interface 1.

   Do not use the VLAN configured for ND MAD for any other purposes.

   Perform this task also on the intermediate device (if any).
5. Return to system view.

   quit
6. Enter Ethernet interface view or interface range view.
   - Enter Ethernet interface view.

     interface *interface-type interface-number*
   - Enter interface range view. Choose one of the following commands:

     interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

     interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

   To assign a range of ports to the ND MAD VLAN, enter interface range view.

   To assign one port to the ND MAD VLAN, enter Ethernet interface view.
7. Assign the port or the range of ports to the ND MAD VLAN.
   - Assign the ports to the VLAN as access ports.

     port access vlan *vlan-id*
   - Assign the ports to the VLAN as trunk ports.

     port trunk permit vlan *vlan-id*
   - Assign the ports to the VLAN as hybrid ports.

     port hybrid vlan *vlan-id* { **tagged** | **untagged** }

   The link type of ND MAD ports can be access, trunk, or hybrid.

   The default link type of a port is access.

   Perform this task also on the intermediate device (if any).
8. Return to system view.

   quit
9. Enter VLAN interface view.

   interface vlan-interface *vlan-interface-id*
10. Assign the interface an IPv6 address.

```
ipv6 address { ipv6-address/prefix-length | ipv6-address
prefix-length }
```

By default, no IPv6 addresses are assigned to any VLAN interfaces.

11. Enable ND MAD.

```
mad nd enable
```

By default, ND MAD is disabled.

Excluding interfaces from the shutdown action upon detection of multi-active collision

About this task

When an IRF fabric transits to the Recovery state, the system automatically excludes the following network interfaces from being shut down:

- IRF physical interfaces.
- Member interfaces of an aggregate interface if the aggregate interface is excluded from being shut down.

You can exclude a network interface from the shutdown action for management or other special purposes. For example:

- Exclude a port from the shutdown action so you can Telnet to the port for managing the device.
- Exclude a VLAN interface and its Layer 2 ports from the shutdown action so you can log in through the VLAN interface.

Restrictions and guidelines

If the Layer 2 ports of a VLAN interface are distributed on multiple member devices, the exclusion operation might introduce IP collision risks. The VLAN interface might be up on both active and inactive IRF fabrics.

Procedure

1. Enter system view.

```
system-view
```

2. Configure a network interface to not shut down when the IRF fabric transits to the Recovery state.

```
mad exclude interface interface-type interface-number
```

By default, all network interfaces on a Recovery-state IRF fabric are shut down, except for the network interfaces automatically excluded by the system.

Recovering an IRF fabric

About this task

For split IRF fabrics, if the active IRF fabric fails before the IRF link is recovered, perform this task on the inactive IRF fabric to recover the inactive IRF fabric. The manual recovery operation brings up all interfaces that were shut down by MAD on the inactive IRF fabric.

Procedure

1. Enter system view.

```
system-view
```

2. Recover the inactive IRF fabric.

```
mad restore
```

Optimizing IRF settings for an IRF fabric

Configuring a member device description

1. Enter system view.

   system-view

2. Configure a description for a member device.

   irf member *member-id* **description** *text*

   By default, no member device description is configured.

Configuring IRF link load sharing mode

About IRF link load sharing mode

On an IRF port, traffic is balanced across its physical links.

You can configure the IRF port to distribute traffic based on any combination of the following criteria:

- IP addresses.
- MAC addresses.

The system displays an error message if a criteria combination is not supported.

The criteria can also be packet types, such as Layer 2, IPv4, and IPv6.

Restrictions and guidelines for configuring IRF link load sharing mode

Configure the IRF link load sharing mode for IRF links in system view or IRF port view:

- In system view, the configuration is global and takes effect on all IRF ports.
- In IRF port view, the configuration is port specific and takes effect only on the specified IRF port.

An IRF port preferentially uses the port-specific load sharing mode. If no port-specific load sharing mode is available, the IRF port uses the global load sharing mode.

Before you configure a port-specific load sharing mode, make sure you have bound a minimum of one physical interface to the IRF port.

Configuring the global load sharing mode

1. Enter system view.

   system-view

2. Configure the global IRF link load sharing mode.

   irf-port global load-sharing mode { **destination-ip** | **destination-mac** | **source-ip** | **source-mac** } *

   By default, packets are distributed based on the load sharing mode automatically selected depending on the packet type.

   If you execute this command multiple times, the most recent configuration takes effect.

Configuring a port-specific load sharing mode

1. Enter system view.

   system-view

2. Enter IRF port view.

   irf-port *member-id/irf-port-number*

3. Configure the port-specific load sharing mode.

irf-port load-sharing mode { **destination-ip** | **destination-mac** | **source-ip** | **source-mac** } *

By default, packets are distributed based on the load sharing mode automatically selected depending on the packet type.

If you execute this command multiple times, the most recent configuration takes effect.

Configuring the IRF bridge MAC address

About this task

The bridge MAC address of a system must be unique on a switched LAN. IRF bridge MAC address identifies an IRF fabric by Layer 2 protocols (for example, LACP) on a switched LAN.

By default, an IRF fabric uses the bridge MAC address of the master as the IRF bridge MAC address. After the master leaves, the IRF bridge MAC address persists for a period of time or permanently depending on the IRF bridge MAC persistence setting. When the IRF bridge MAC persistence timer expires, the IRF fabric uses the bridge MAC address of the current master as the IRF bridge MAC address.

If IRF fabric merge occurs, IRF determines the IRF bridge MAC address of the merged IRF fabric as follows:

1. When IRF fabrics merge, IRF ignores the IRF bridge MAC addresses and checks the bridge MAC address of each member device in the IRF fabrics. IRF merge fails if any two member devices have the same bridge MAC address.
2. After IRF fabrics merge, the merged IRF fabric uses the bridge MAC address of the merging IRF fabric that won the master election as the IRF bridge MAC address.

Restrictions and guidelines

△ **CAUTION:**

Bridge MAC address change will cause transient traffic disruption.

When you configure IRF bridge MAC persistence, follow these restrictions and guidelines:

- If ARP MAD or ND MAD is used with the spanning tree feature, you must disable IRF bridge MAC persistence by using the **undo irf mac-address persistent** command.
- If the IRF fabric has multichassis aggregate links, do not use the **undo irf mac-address persistent** command. Use of this command might cause traffic disruption.

An IRF bridge MAC change might cause the MAC addresses of logical interfaces such as VLAN interfaces to change, resulting in forwarding issues. To avoid this situation, configure IRF bridge MAC persistence to have the existing IRF bridge MAC address persists for a while or permanently after the address owner leaves the IRF fabric.

Configuring IRF bridge MAC persistence

1. Enter system view.

   system-view

2. Configure IRF bridge MAC persistence.
   - Retain the bridge MAC address permanently even if the address owner has left the fabric.

     irf mac-address persistent always
   - Retain the bridge MAC address for 6 minutes after the address owner leaves the fabric.

     irf mac-address persistent timer
   - Change the bridge MAC address as soon as the address owner leaves the fabric.

     undo irf mac-address persistent

By default, the IRF bridge MAC address remains unchanged for 6 minutes after the address owner leaves.

The **`irf mac-address persistent timer`** command avoids unnecessary bridge MAC address changes caused by device reboot, transient link failure, or purposeful link disconnection.

Enabling software auto-update for software image synchronization

About this task

The software auto-update feature automatically synchronizes the current software images of the master to devices that are attempting to join the IRF fabric.

To join an IRF fabric, a device must use the same software images as the master in the fabric.

When you add a device to the IRF fabric, software auto-update compares the startup software images of the device with the current software images of the IRF master. If the two sets of images are different, the device automatically performs the following operations:

1. Downloads the current software images of the master.
2. Sets the downloaded images as its main startup software images.
3. Reboots with the new software images to rejoin the IRF fabric.

You must manually update the new device with the software images running on the IRF fabric if software auto-update is disabled.

Restrictions and guidelines

To ensure a successful software auto-update in a multi-user environment, prevent anyone from rebooting member devices during the auto-update process. To inform administrators of the auto-update status, configure the information center to output the status messages to configuration terminals (see *Network Management and Monitoring Configuration Guide*).

Make sure the device you are adding to the IRF fabric has sufficient storage space for the new software images.

If sufficient storage space is not available, the device automatically deletes the current software images. If the reclaimed space is still insufficient, the device cannot complete the auto-update. You must reboot the device, and then access the Bootware menu to delete files.

Procedure

1. Enter system view.

```
system-view
```

2. Enable software auto-update.

```
irf auto-update enable
```

   By default, software auto-update is enabled.

Removing an expansion interface card that has IRF physical interfaces

To remove an expansion interface card that provides IRF physical interfaces:

1. Perform one of the following tasks to eliminate temporary packet loss:
   - Remove cables from the IRF physical interfaces on the card.
   - Shut down the IRF physical interfaces on the card by using the **`shutdown`** command.
2. Remove the card.

Replacing an expansion interface card that has IRF physical interfaces

Replacing the old card with a different model replacement card

1. Shut down the IRF physical interfaces on the old card by using the **shutdown** command.
2. Remove the IRF port bindings that contain the physical interfaces.
3. Remove the old card, and then install the replacement card.
4. Verify that the replacement card has been correctly installed by using the **display device** command.
5. Reconfigure the IRF port bindings, as described in "Binding physical interfaces to IRF ports."
6. Activate the IRF port settings by using the **irf-port-configuration active** command.

   You can skip this step if the IRF port is in UP state when you add bindings.

Replacing the old card with the same model replacement card

1. Shut down the IRF physical interfaces on the old card by using the **shutdown** command.
2. Remove the old card, and then install the replacement card.
3. Verify that the replacement card has been correctly installed by using the **display device** command.

   Bring up the physical interfaces by using the **undo shutdown** command after the interface card completes startup.

Display and maintenance commands for IRF

Execute **display** commands in any view.

Task	Command
Display information about all IRF members.	**display irf**
Display the IRF fabric topology.	**display irf topology**
Display IRF link information.	**display irf link**
Display IRF configuration.	**display irf configuration**
Display the load sharing mode for IRF links.	**display irf-port load-sharing mode** [**irf-port** [*member-id/irf-port-number*]]
Display MAD configuration.	**display mad** [**verbose**]

Contents

Configuring contexts

About contexts

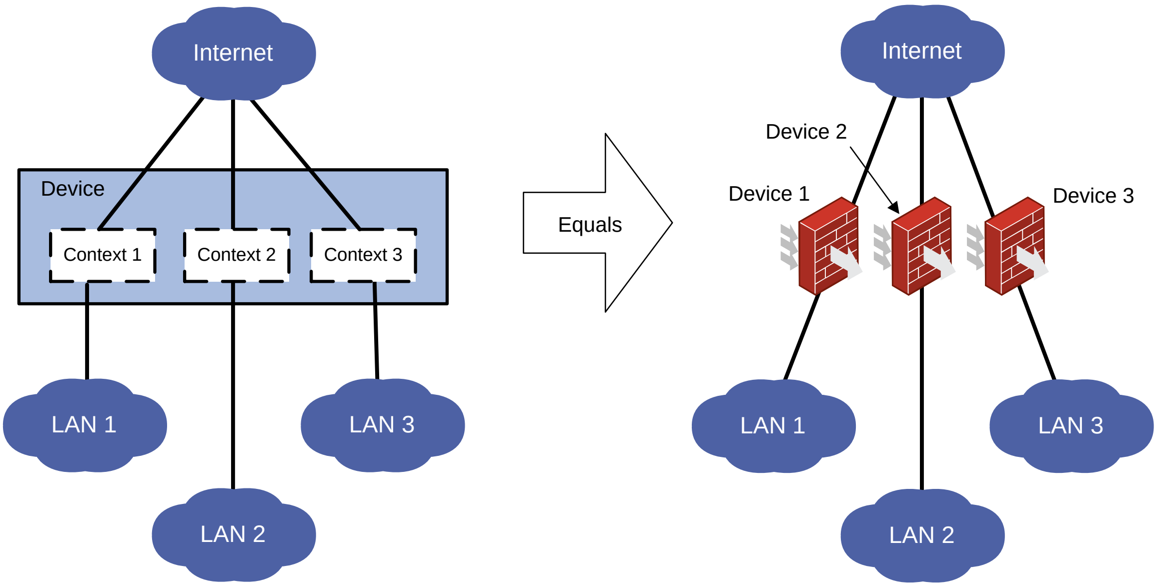
A physical device or an IRF fabric can be virtualized into multiple logical devices called contexts. Each context is assigned separate hardware and software resources, and operates independently of other contexts. From the user's perspective, a context is a standalone device.

Context applications

With context technology, you can meet device requirements from different branches or companies by using a single physical device.

As shown in Figure 1, LAN 1, LAN 2, and LAN 3 are connected to the Internet through the same device. To provide secure access services for the three LANs, you can deploy a single physical device and configure a context for each LAN on the device. The administrator of each LAN can only log in to and manage its own context without affecting other LANs. This has the same effect as deploying a separate device for each LAN.

Figure 1 Network diagram



Default context and non-default contexts

A device supporting contexts is considered to be a context. This context is called the default context (for example, Device in Figure 1). The default context always uses the name **Admin** and the ID 1. You cannot delete it or change its name or ID.

When you log in to the physical device, you are logged in to the default context. On the default context, you can perform the following tasks:

- Manage the entire physical device.
- Create and delete non-default contexts (for example, Context 1, Context 2, and Context 3 in Figure 1).
- Assign resources to non-default contexts. These resources include CPU resources, disk spaces, memory spaces, interfaces, and VLANs.

Administrators of non-default contexts can only use resources assigned to their own contexts. They cannot use free resources or create other contexts. Resources that are not assigned to any non-default context belong to the default context.

A non-default context does not support packet capture on shared interfaces. For more information about packet capture, see *Network Management and Monitoring Configuration Guide*.

Restrictions: Hardware compatibility with context configuration

Models	Context compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080	Yes
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: Context configuration

All commands in this chapter are supported on the default context. On a non-default context, only the **`display context interface`**, **`display context reboot`**, and **`reset context reboot`** commands are supported.

DPI services on non-default contexts use the DPI engine on the default context for packet matching. Creating, deleting, stopping, or restarting a non-default context re-activates the DPI engine on the default context. Before the DPI engine on the default context operates correctly, no contexts can provide DPI services.

Context tasks at a glance

To configure contexts, perform the following tasks:

1. Creating contexts
2. (Optional.) Assigning interfaces and VLANs to a context
   - Assigning interfaces to a context
   - Assigning VLANs to a context
3. (Optional.) Limiting resource use for a context

   - Setting the outbound throughput threshold
   - Setting the maximum number of security policy rules
   - Setting the maximum number of concurrent unicast sessions
   - Setting the upper limit of the session establishment rate
   - Setting the maximum number of SSL VPN users
4. Starting a context
5. (Optional.) Assigning CPU, disk, and memory resources to a context
   - Specifying a CPU weight for a context
   - Specifying a memory space percentage for a context
6. Accessing a context
7. (Optional.) Configuring inbound rate limiting for contexts
8. (Optional.) Setting the CPU usage threshold per CPU core for attack prevention
9. (Optional.) Archiving log messages for contexts
10. (Optional.) Configuring a context to support inter-VPC connectivity

Creating contexts

Restrictions and guidelines

When you create a context, you can assign it the VLAN-unshared attribute as required.

- A context with the VLAN-unshared attribute has its own VLAN resources (VLAN 2 through VLAN 4094). It does not share VLAN resources with any other context. To create VLANs for the context, log in to the context and use the **`vlan`** command. VLAN 1 is system defined. You cannot create or delete VLAN 1.
- All contexts without the VLAN-unshared attribute share the same VLAN resources (VLAN 1 through VLAN 4094). You can create VLANs on the default context and use the **`allocate vlan`** command to assign VLANs to the contexts. A VLAN can be assigned only to one context. VLAN 1 is system defined. It belongs to the default context. You cannot assign it to a non-default context.

  For a context without the VLAN-unshared attribute, you cannot perform the following tasks:

  - Change the link mode of Layer 3 Ethernet interfaces on the context to Layer 2.
  - Assign a Layer 2 Ethernet interface exclusively to the context.

Procedure

1. Enter system view.

   `system-view`

2. Create a context and enter context view.

   `context` *`context-name`* `[` **`id`** *`context-id`* `]` `[` **`vlan-unshared`** `]`

   By default, a default context exists. The context name is **Admin** and the context ID is 1.

3. Configure a description for the context.

   `description` *`text`*

   By default, the default context uses the description **DefaultContext**, and a non-default context does not have a description.

Assigning interfaces and VLANs to a context

Assigning interfaces to a context

About this task

By default, all interfaces belong to the default context. A non-default context cannot use any interfaces. To enable a non-default context to communicate, you must assign it interfaces.

You can assign interfaces to contexts in exclusive or shared mode:

- **Exclusive mode**—You assign an interface exclusively to a context, and only the context can use the interface. The administrator of the context can see the interface and use all commands supported on the interface.
- **Shared mode**—You assign an interface to multiple contexts in shared mode, and the system creates a virtual interface for each context. The virtual interfaces use the same name as the physical interface but have different MAC addresses and IP addresses. They forward and receive packets through the physical interface. The shared mode improves interface usage.

  You can see the physical interface and perform all commands supported on the interface from the default context. The administrator of a context can only see the context's virtual interface and use the shutdown, description, and network- and security-related commands.

Restrictions and guidelines

Do not assign IRF physical interfaces to a non-default context.

Do not assign member interfaces of an aggregate interface to a non-default context.

If a subinterface of a Layer 3 interface is a member interface of a Reth interface, do not assign the Layer 3 interface to a non-default context.

A logical interface (for example, a subinterface or aggregate interface) can be assigned to a context only in shared mode.

After assigning a subinterface to a context, you cannot assign its primary interface to a context. After assigning a primary interface to a context, you cannot assign its subinterfaces to a context.

After assigning an interface to contexts in shared mode, you cannot assign the interface to contexts in exclusive mode before reclaiming the interface.

For non-default contexts to communicate with each other, you must assign physical or logical interfaces to the non-default contexts in shared mode on the default context.

Procedure

1. Enter system view.

   `system-view`

2. Enter context view.

   `context` *`context-name`*

3. Assign interfaces to the context.
   - Assign individual interfaces to the context.

     `allocate interface` `{` *`interface-type interface-number`* `}&<1-24> [` **`share`** `]`
   - Assign a range of interfaces to the context.

     `allocate interface` *`interface-type interface-number1`* **`to`** *`interface-type interface-number2`* `[` **`share`** `]`

   By default, all interfaces belong to the default context. A non-default context cannot use any interfaces.

Assigning VLANs to a context

Restrictions and guidelines

For contexts without the VLAN-unshared attribute, you can only assign VLANs to them and cannot use the **vlan** command to create VLANs for them. Before the assignment, you must create the VLANs on the default context. A VLAN can be assigned only to one context.

You cannot assign the following VLANs to a context without the VLAN-unshared attribute:

- VLAN 1.
- Default VLANs of interfaces.
- VLANs for which you have created VLAN interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter context view.

   context *context-name*

3. Assign VLANs to the context.
   - Assign individual VLANs to the context.

     allocate vlan *vlan-id*&<1-24>
   - Assign a range of VLANs to the context.

     allocate vlan *vlan-id1* **to** *vlan-id2*

   By default, no VLANs are assigned to a context.

Limiting resource use for a context

Setting the outbound throughput threshold

About this task

This feature limits the outbound throughput for a context to prevent it from occupying too many shared resources on a security engine.

Procedure

1. Enter system view.

   system-view

2. Enter context view.

   context *context-name*

3. Set the outbound throughput threshold on the context.

   capability throughput { **kbps** | **pps** } *value*

   By default, the outbound throughput is not limited on a context.

Setting the maximum number of security policy rules

About this task

A large number of rules occupy too much memory, affecting other features on the context. This feature limits the number of security policy rules for a context. When the maximum number is reached, you cannot add new rules. For information about security policies, see *Security Configuration Guide*.

Restrictions and guidelines

If the maximum number you set is smaller than the number of existing security policy rules, this setting takes effect. The context does not delete extra existing security policy rules and allows new security policy rules to be created only when the number of security policy rules drops below the maximum number.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter context view.

   ```
   context context-name
   ```

3. Set the maximum number of security policy rules.

   ```
   capability security-policy-rule maximum max-value
   ```

   By default, the number of security policy rules is not limited for a context.

Setting the maximum number of concurrent unicast sessions

About this task

A large number of sessions occupy too much memory, affecting establishment of sessions on other contexts. This feature limits the number of concurrent unicast sessions for a context. When the maximum number is reached, you cannot establish additional unicast sessions.

This feature does not affect local traffic, such as FTP traffic, Telnet traffic, SSH traffic, HTTP traffic, and HTTP-based load balancing traffic.

Restrictions and guidelines

If the maximum number you set is smaller than the number of existing concurrent unicast sessions, this setting takes effect. The context does not delete extra existing concurrent unicast sessions and allows new unicast sessions to be created only when the number of concurrent unicast sessions drops below the maximum number.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter context view.

   ```
   context context-name
   ```

3. Set the maximum number of concurrent unicast sessions.

   ```
   capability session maximum max-number
   ```

   By default, the number of concurrent unicast sessions is not limited.

Setting the upper limit of the session establishment rate

About this task

Establishing sessions too frequently consumes too much CPU resources. If a context establishes sessions too frequently, other contexts will not be able to establish sessions. This feature limits the number of sessions that can be established per second for a context. When the upper limit is reached for a context, no additional sessions can be established.

This feature does not affect local traffic, such as FTP traffic, Telnet traffic, SSH traffic, HTTP traffic, and HTTP-based load balancing traffic.

Procedure

1. Enter system view.

   `system-view`

2. Enter context view.

   `context` *`context-name`*

3. Set the upper limit of the session establishment rate.

   `capability session rate` *`max-value`*

   By default, the session establishment rate is not limited for a context.

Setting the maximum number of SSL VPN users

About this task

The maximum number of SSL VPN users allowed on the device is license-restricted. If the number of logged-in users in a context already reaches the total upper limit, other contexts cannot accept login requests. This feature limits the number of SSL VPN users that can log in to a context. When the maximum number is reached, the context will reject the login requests of new SSL VPN users.

Restrictions and guidelines

If the maximum number you set is smaller than the number of SSL VPN users that already have logged in to a context, this setting takes effect. The context does not log out the currently logged-in users and allows new users to log in only when the number of the logged-in users drops below the maximum number.

Procedure

1. Enter system view.

   `system-view`

2. Enter context view.

   `context` *`context-name`*

3. Set the maximum number of SSL VPN users.

   `capability sslvpn-user maximum` *`max-number`*

   By default, the number of SSL VPN users is not limited for a context. The number is determined by the usage of the SSL VPN licenses installed on the device.

Starting a context

About this task

You must perform this task to initialize a newly created context. You can configure a context only after it is started.

When a context starts, the device examines whether requirements for starting the context are met to ensure status consistency between the master and backup contexts and correct operation of the context.

In an IRF fabric, memory insufficiency might occur during master and backup switchover or configuration recovery. Some contexts will stay in updating or inactive status because of status inconsistency between master and backup processes although they can process services. Use the **`context start force`** command to forcibly start these contexts after the memory becomes sufficient. The device will recover the abnormal context processes without service interruption.

Restrictions and guidelines

Before using the **`context start force`** command, you can use the following commands to display the context running information:

- **`display context`**
- **`display system internal context configuration-status`**
- **`display system internal`** *`context id`* **`context-id running-status`**

Procedure

1. Enter system view.

   `system-view`
2. Enter context view.

   `context` *`context-name`*
3. Start the context.

   `context start` `[` **`force`** `]`

   By default, a context is not started.

Assigning CPU, disk, and memory resources to a context

About CPU, disk, and memory resources assignment

When you assign a context to a security engine group, the system automatically assigns CPU, disk space, and memory space resources on the security engines to the context. All contexts residing on the same security engine share and compete for the engine's free CPU, disk, and memory resources. To prevent one context from occupying too many resources, assign CPU, disk space resources, and memory space resources to the contexts. To assign resources to a context, specify a CPU weight, disk space percentage, and memory space percentage for the context.

Specifying a CPU weight for a context

About this task

When the CPU resources on a security engine cannot meet the processing requirements from contexts, the system allocates CPU resources on the engine as follows:

1. Identifies the CPU weights of all contexts on the engine.
2. Calculates the percentage of each context's CPU weight among the CPU weights of all contexts.
3. Allocates CPU resources to contexts based on their CPU weight percentages.

For example, three contexts share the same CPU. You can assign a weight of 2 to the key context and a weight of 1 to each of the other two contexts. When the system is running out of CPU resources, the key context can use approximately two times of the CPU resources that each of the other two contexts can use.

Procedure

1. Enter system view.

   `system-view`
2. Enter context view.

   `context` *`context-name`*

3. Specify a CPU weight for the context.

   limit-resource cpu weight *weight-value*

   By default, each context has a CPU weight of 10.

Specifying a memory space percentage for a context

Restrictions and guidelines

To prevent a context from start failures because of memory space insufficiency, specify a memory space percentage for the context after the context has started correctly.

After the context starts, make sure the configured memory space limit meets the memory space needs of the services provided by the context.

Procedure

1. Enter system view.

   system-view

2. Enter context view.

   context *context-name*

3. Display the amount of memory space that has been used by the context.

   display context resource memory

4. Specify a memory space percentage for the context.

   limit-resource memory slot *slot-number* **cpu** *cpu-number* **ratio** *limit-ratio*

   By default, all contexts share the memory space in the system. A context can use all free memory space.

Accessing a context

About this task

From the system view of the default context, you can log in to a non-default context and enter the context's user view.

You can also access a context by using Telnet and SSH.

Procedure

1. Enter system view.

   system-view

2. Log in to a context.

   switchto context *context-name*

   To return to the default context, use the **quit** command.

Configuring inbound rate limiting for contexts

Configuring inbound broadcast rate limiting

About this task

Inbound broadcast rate limiting controls the rates of incoming broadcast packets on contexts. This feature can prevent a context from using too many resources and degrading the service processing capabilities of other contexts.

Inbound broadcast rate limiting uses the following types of limits:

- **Per-context broadcast rate limit**—Limit on the rate of incoming broadcast packets on a single context.
- **Total broadcast rate limit**—Limit on the total rate of incoming broadcast packets on all contexts.

When both a per-context broadcast rate limit and the total broadcast rate limit are reached, the device drops subsequent broadcast packets that arrive at the context.

Restrictions and guidelines

This feature applies only to inbound broadcast packets.

This feature takes effect only on active contexts that share interfaces with other contexts.

Setting the total inbound broadcast rate limit to 0 disables the inbound broadcast rate limiting feature.

Procedure

1. Enter system view.
   system-view
2. Set the total inbound broadcast rate limit.
   context-capability inbound broadcast total pps *threshold*
   The default setting varies by device model. For more information, see the command reference.
3. Set the inbound broadcast rate limit for the default context.
   context-capability inbound broadcast single pps *threshold*
   By default, the inbound broadcast rate limit for the default context is the total rate limit divided by the number of active contexts that share interfaces with other contexts.
4. Enter the view of a non-default context.
   context *context-name*
5. Set the inbound broadcast rate limit for the context.
   context-capability inbound broadcast single pps *threshold*
   By default, the inbound broadcast rate limit for a non-default context is the total rate limit divided by the number of active contexts that share interfaces with other contexts.

Configuring inbound multicast rate limiting

About this task

Inbound multicast rate limiting controls the rates of incoming multicast packets on contexts. This feature can prevent a context from using too many resources and degrading the service processing capabilities of other contexts.

Inbound multicast rate limiting uses the following types of limits:

- **Per-context multicast rate limit**—Limit on the rate of incoming multicast packets on a single context.
- **Total multicast rate limit**—Limit on the total rate of incoming multicast packets on all contexts.

When both a per-context inbound multicast rate limit and the total inbound multicast rate limit are reached, the device drops subsequent multicast packets that arrive at the context.

Restrictions and guidelines

This feature applies only to inbound multicast packets.

This feature takes effect only on active contexts that share interfaces with other contexts.

Setting the total inbound multicast rate limit to 0 disables the inbound multicast rate limiting feature.

Procedure

1. Enter system view.
   system-view
2. Set the total inbound multicast rate limit.
   context-capability inbound multicast total pps *threshold*

   The default setting varies by device model. For more information, see the command reference.
3. Set the inbound multicast rate limit for the default context.
   context-capability inbound multicast single pps *threshold*

   By default, the inbound multicast rate limit for the default context is the total rate limit divided by the number of active contexts that share interfaces with other contexts.
4. Enter the view of a non-default context.
   context *context-name*
5. Set the multicast rate limit for the context.
   context-capability inbound multicast single pps *threshold*

   By default, the inbound multicast rate limit for a non-default context is the total rate limit divided by the number of active contexts that share interfaces with other contexts.

Enabling logging for packets dropped because of rate limiting

About this task

This logging feature generates and sends a log message to the information center when an incoming packet is dropped because of rate limiting on contexts. For more information about how the information center manages log messages, see information center configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.
   system-view
2. Enable logging for incoming packets dropped because of rate limiting on contexts.
   context-capability inbound drop-logging enable

   By default, logging is disabled for incoming packets that are dropped because of rate limiting on contexts.

Setting the CPU usage threshold per CPU core for attack prevention

About this task

Perform this task to set the CPU usage threshold per CPU core for all inbound packets from all contexts. The specified threshold applies to all inbound packets, including broadcast, unicast, and multicast packets.

If the shared queue in the driver is full when the total usage of a CPU core reaches the specified threshold, the system determines that an attack risk is present. Then, it takes the attack prevention action configured by using the **`attack-defense cpu-core action`** command until the attack risk is eliminated. For more information about the **`attack-defense cpu-core action`** command, see attack detection and prevention commands in *Security Command Reference*.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280	Yes
NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the CPU usage threshold per CPU core for all inbound packets from all contexts for attack prevention.

   ```
   context-capability inbound unicast total cpu-usage threshold
   ```

   By default, the CPU usage threshold per CPU core is 95%.

Archiving log messages for contexts

About this task

This feature archives all files in the **logfile** directory and **diagfile** directory.

Procedure

To archive log messages for contexts, execute this command in the user view:

```
tar context [ name context-name ] log file filename
```

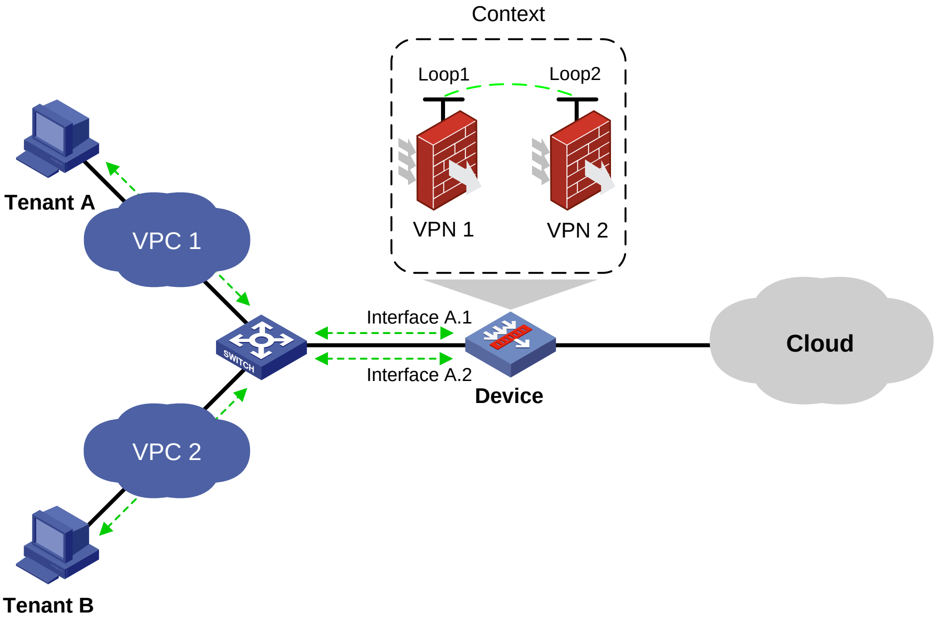
Configuring a context to support inter-VPC connectivity

About this task

In a virtual private cloud (VPC) scenario, the following tasks are required for tenant isolation and inter-VPC connectivity:

- Configure VPN instances, each of which corresponds to a VPC, thus isolating traffic of tenants in different VPCs.
- Configure static routes for the VPN instances to implement interconnectivity of VPCs. In the static routes, loopback interfaces are specified as the output interfaces.

Figure 2 Inter-VPC connectivity



Restrictions and guidelines

If a context is configured to support inter-VPC connectivity, make sure you configure only compatible related services on the context. Compatibility of related services with inter-VPC connectivity is as follows:

- Packet filtering services such as security policy are supported. However, services that will change the IP address of packets such as NAT and load balancing are not supported.
- Static routing is supported.

As a best practice, use subinterfaces for the device to communicate with tenants in VPCs.

Procedure

1. Enter system view.
   `system-view`
2. Create VPN instances and associate an interface with each VPN instance.
   For more information, see VPN instance configuration in *VPN Instance Configuration Guide*.
3. Configure static routes for the VPN instances and specify loopback interfaces as the output interfaces for the static routes.
   For more information, see static routing configuration and IPv6 static routing configuration in *Layer 3—IP Routing Configuration Guide*.

Display and maintenance commands for contexts

Execute **`display`** commands in any view on the default context.

Execute **`reset`** commands in user view on the default context.

Task	Command
Display contexts.	**display context** [**name** *context-name*] [**verbose**]
Display usage of allocable service resources for contexts.	**display context** [**name** *context-name*] **capability** [**security-policy** | **session** [**slot** *slot-number*] | **sslvpn-user**]
Display the inbound broadcast rate limit statitiscs for a context.	**display context name** *context-name* **capability inbound broadcast slot** *slot-number*
Display the inbound multicast rate limit statitiscs for a context.	**display context name** *context-name* **capability inbound multicast slot** *slot-number*
Display attack prevention statistics for CPU cores.	**display capability inbound unicast slot** *slot-number*
Display context configuration information.	**display context** [**name** *context-name*] **configuration** [**file** *filename*]
Display interfaces assigned to contexts.	**display context** [**name** *context-name*] **interface**
Display CPU, disk space, and memory usage of contexts.	**display context** [**name** *context-name*] **resource** [**cpu** | **disk** | **memory**] [**slot** *slot-number* **cpu** *cpu-number*]
Display resource statistics for contexts.	**display context** [**name** *context-name*] **statistics** [**file** *filename*]
Display non-default context reboot information.	**display context name** *context-name* **reboot** *show-number* [*offset*]
Display the number of online SSL VPN users on all contexts.	**display context online-users sslvpn**
Display VLAN lists for contexts.	**display context** [**name** *context-name*] **vlan**
Clear the inbound broadcast rate limit statistics for a context.	**reset context name** *context-name* **capability inbound broadcast slot** *slot-number*
Clear the inbound multicast rate limit statistics for a context.	**reset context name** *context-name* **capability inbound multicast slot** *slot-number*
Clear non-default context reboot information.	**reset context** [**name** *context-name*] **reboot**

Execute **display** commands in any view and **reset** commands in user view on a non-default context:

Task	Command
Display interfaces assigned to the context.	**display context** [**name** *context-name*] **interface**
Display reboot information about the current context.	**display context reboot** *show-number* [*offset*]
Clear reboot information about the current context.	**reset context reboot**

Context configuration examples

Example: Configuring contexts

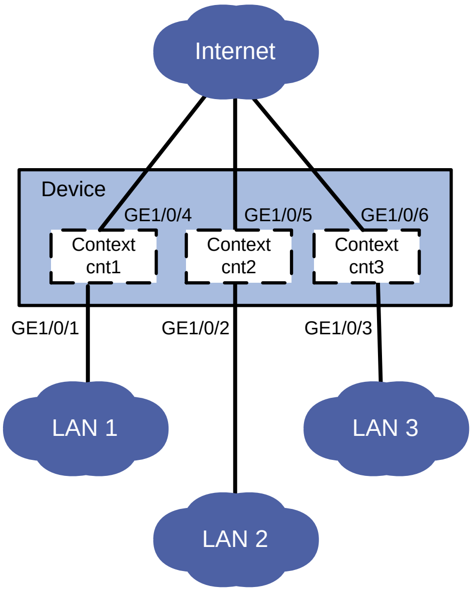
Network configuration

As shown in Figure 3, LAN 1, LAN 2, and LAN 3 use 192.168.1.0/24, 192.168.2.0/24, and 192.168.3.0/24, respectively.

Configure contexts for the LANs as follows:

- Configure context **cnt1** for LAN 1. Assign 60% disk space and 60% memory space to the context and set the CPU weight to 8.
- Configure context **cnt2** for LAN 2. Leave the context to use the default amount of disk space and the default amount of memory space.
- Configure context **cnt3** for LAN 3. Set the CPU weight to 2.

Figure 3 Network diagram



Procedure

1. Configure context **cnt1** for LAN 1.

   # Create a context named **cnt1** and configure a description for it.

   ```
   <Device> system-view
   [Device] context cnt1
   [Device-context-2-cnt1] description context-1
   ```

   # Set both the disk space percentage and memory space percentage to 60% and set the CPU weight to 8 for the context.

   ```
   [Device-context-2-cnt1] limit-resource disk slot 1 cpu 0 ratio 60
   [Device-context-2-cnt1] limit-resource memory slot 1 cpu 0 ratio 60
   [Device-context-2-cnt1] limit-resource cpu weight 8
   ```

   # Assign GigabitEthernet 1/0/1 and GigabitEthernet 1/0/4 to the context.

   ```
   [Device-context-2-cnt1] allocate interface gigabitethernet 1/0/1 gigabitethernet
   1/0/4
   Configuration of the interfaces will be lost. Continue? [Y/N]:y
   ```

# Start the context.

```
[Device-context-2-cnt1] context start
It will take some time to start the context...
Context started successfully.
[Device-context-2-cnt1] quit
```

# Log in to the context from the default context.

```
[Device] switchto context cnt1
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<NSFOCUS> system-view
```

# Configure Telnet login to enable remote context management. (Details not shown. For more information about Telnet login configuration, see login management in *Fundamentals Configuration Guide.*)

# Change the device name to **cnt1** for easy identification of the context.

```
[NSFOCUS] sysname cnt1
```

# Assign IP address 192.168.1.251/24 to GigabitEthernet 1/0/1.

```
[cnt1] interface gigabitethernet 1/0/1
[cnt1-GigabitEthernet1/0/1] ip address 192.168.1.251 24
```

# Return to the default context.

```
[cnt1-GigabitEthernet1/0/1] return
<cnt1> quit
[Device]
```

2. Configure context **cnt2** for LAN 2.

# Create a context named **cnt2** and configure a description for it.

```
[Device] context cnt2
[Device-context-3-cnt2] description context-2
```

# Assign GigabitEthernet 1/0/2 and GigabitEthernet 1/0/5 to the context.

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/2 gigabitethernet
1/0/5
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

# Start the context.

```
[Device-context-3-cnt2] context start
It will take some time to start the context...
Context started successfully.
[Device-context-3-cnt2] quit
```

# Log in to the context from the default context.

```
[Device] switchto context cnt2
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS.   All rights reserved.                    *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<NSFOCUS> system-view
```

# Configure Telnet login to enable remote context management. (Details not shown. For more information about Telnet login configuration, see login management in *Fundamentals Configuration Guide.*)

# Change the device name to **cnt2** for easy identification of the context.

```
[NSFOCUS] sysname cnt2
```

# Assign IP address 192.168.2.251/24 to GigabitEthernet 1/0/2.

```
[cnt2] interface gigabitethernet 1/0/2
[cnt2-GigabitEthernet1/0/2] ip address 192.168.2.251 24
```

# Return to the default context.

```
[cnt2-GigabitEthernet1/0/2] return
<cnt2> quit
[Device]
```

3. Configure context **cnt3** for LAN 3.

# Create a context named **cnt3** and configure a description for it.

```
[Device] context cnt3
[Device-context-4-cnt3] description context-3
```

# Set the CPU weight to 2 for the context.

```
[Device-context-4-cnt3] limit-resource cpu weight 2
```

# Assign GigabitEthernet 1/0/3 and GigabitEthernet 1/0/6 to the context.

```
[Device-context-4-cnt3] allocate interface gigabitethernet 1/0/3 gigabitethernet
1/0/6
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

# Start the context.

```
[Device-context-4-cnt3] context start
It will take some time to start the context...
Context started successfully.
[Device-context-4-cnt3] quit
```

# Log in to the context from the default context.

```
[Device] switchto context cnt3
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<NSFOCUS> system-view
```

# Configure Telnet login to enable remote context management. (Details not shown. For more information about Telnet login configuration, see login management in *Fundamentals Configuration Guide.*)

# Change the context name to **cnt3** for easy identification of the context.

```
[NSFOCUS] sysname cnt3
```

# Assign IP address 192.168.3.251/24 to GigabitEthernet 1/0/3.

```
[cnt3] interface gigabitethernet 1/0/3
[cnt3-GigabitEthernet1/0/3] ip address 192.168.3.251 24
```

# Return to the default context.

```
[cnt3-GigabitEthernet1/0/3] return
<cnt3> quit
[Device]
```

Verifying the configuration

# Verify that the device has four contexts and all contexts are in active state.

```
[Device] display context
ID      Name             Status       Description
1       Admin            active       DefaultContext
2       cnt1             active       context-1
3       cnt2             active       context-2
4       cnt3             active       context-3
```

# Telnet to context **cnt1** and view the running configuration on the context.

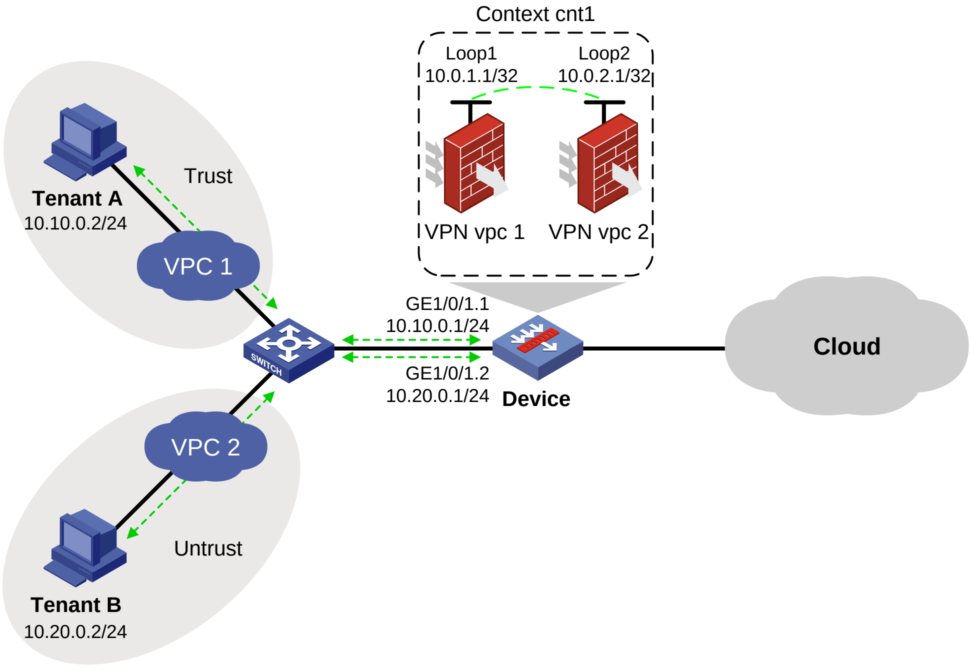
```
C:\> telnet 192.168.1.251
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<cnt1> display current-configuration
...
```

Example: Configuring a context to support inter-VPC connectivity

Network configuration

As shown in Figure 4, configure a context to provide security data access to the public cloud for tenants in VPCs as well as interconnectivity between the VPCs.

Figure 4 Network diagram



Procedure

1. Create a context.

   # Create context **cnt1**, and assign CPU, memory, disk space, and interface resources to the context. For more information, see "Example: Configuring contexts." (Details not shown.)

2. Log in to context **cnt1** from the default context.

   ```
   <Device> system-view
   [Device] switchto context cnt1
   ******************************************************************************
   * Copyright (c) 2004-2020 NSFOCUS. All rights reserved.                      *
   * Without the owner's prior written consent,                                 *
   * no decompiling or reverse-engineering shall be allowed.                    *
   ******************************************************************************

   <NSFOCUS> system-view
   ```

   # Configure Telnet to ensure that users can log in to the context. For more information, see login management configuration in *Fundamentals Configuration Guide*. (Details not shown.)

   # Set the name of context **cnt1** to **cnt1**.

   ```
   [NSFOCUS] sysname cnt1
   ```

3. Create VPN instances, and associate interfaces with the VPN instances.

   # Create VPN instances, configure loopback interfaces and Ethernet subinterfaces, and configure Ethernet subinterfaces to terminate VLAN-tagged packets with the specified outermost VLAN IDs.

   ```
   [cnt1] ip vpn-instance vpc1
   [cnt1-vpn-instance-vpc1] quit
   [cnt1] ip vpn-instance vpc2
   [cnt1-vpn-instance-vpc2] quit
   [cnt1] interface loopback 1
   [cnt1-LoopBack1] ip binding vpn-instance vpc1
   [cnt1-LoopBack1] ip address 10.0.1.1 255.255.255.255
   [cnt1-LoopBack1] quit
   [cnt1] interface loopback 2
   [cnt1-LoopBack2] ip binding vpn-instance vpc2
   [cnt1-LoopBack2] ip address 10.0.2.1 255.255.255.255
   [cnt1-LoopBack2] quit
   [cnt1] interface gigabitethernet 1/0/1.1
   [cnt1-GigabitEthernet 1/0/1.1] ip binding vpn-instance vpc1
   [cnt1-GigabitEthernet 1/0/1.1] ip address 10.10.0.1 255.255.255.0
   [cnt1-GigabitEthernet 1/0/1.1] vlan-type dot1q vid 10
   [cnt1-GigabitEthernet 1/0/1.1] quit
   [cnt1] interface gigabitethernet 1/0/1.2
   [cnt1-GigabitEthernet 1/0/1.2] ip binding vpn-instance vpc2
   [cnt1-GigabitEthernet 1/0/1.2] ip address 10.20.0.1 255.255.255.0
   [cnt1-GigabitEthernet 1/0/1.2] vlan-type dot1q vid 20
   [cnt1-GigabitEthernet 1/0/1.2] quit
   ```

4. Configure settings for routing.

   This example configures static routes to ensure that the VPCs can reach each other.

   ```
   [cnt1] ip route-static vpn-instance vpc1 10.20.0.0 24 loopback1 10.0.2.1
   [cnt1] ip route-static vpn-instance vpc2 10.10.0.0 24 loopback2 10.0.1.1
   ```

5. **Configure security zones.**

   # Add loopback interfaces and Ethernet subinterfaces to security zones.

```
[cnt1] security-zone name trust
[cnt1-security-zone-Trust] import interface loopback1
[cnt1-security-zone-Trust] import interface gigabitethernet 1/0/1.1
[cnt1-security-zone-Trust] quit
[cnt1] security-zone name untrust
[cnt1-security-zone-Untrust] import interface loopback2
[cnt1-security-zone-Untrust] import interface gigabitethernet 1/0/1.2
[cnt1-security-zone-Untrust] quit
```

6. **Configure a security policy to ensure that traffic can be forwarded between the VPCs.**

```
[cnt1] security-policy ip
[cnt1-security-policy-ip] rule name vpc1
[cnt1-security-policy-ip-0-vpc1] action pass
[cnt1-security-policy-ip-0-vpc1] vrf vpc1
[cnt1-security-policy-ip-0-vpc1] source-zone trust
[cnt1-security-policy-ip-0-vpc1] destination-zone untrust
[cnt1-security-policy-ip-0-vpc1] quit
[cnt1-security-policy-ip] rule name vpc2
[cnt1-security-policy-ip-1-vpc2] action pass
[cnt1-security-policy-ip-1-vpc2] vrf vpc2
[cnt1-security-policy-ip-1-vpc2] source-zone untrust
[cnt1-security-policy-ip-1-vpc2] destination-zone trust
[cnt1-security-policy-ip-1-vpc2] quit
[cnt1-security-policy-ip] quit
```

Verifying the configuration

# Ping Tenant B from Tenant A.

```
C:\> ping 10.20.0.2

Pinging 10.20.0.2 with 32 bytes of data:

Reply from 10.20.0.2: bytes=32 time=19ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254

Ping statistics for 10.20.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 19ms, Average = 4ms
```

The output shows that Tenant B can ping Tenant A.

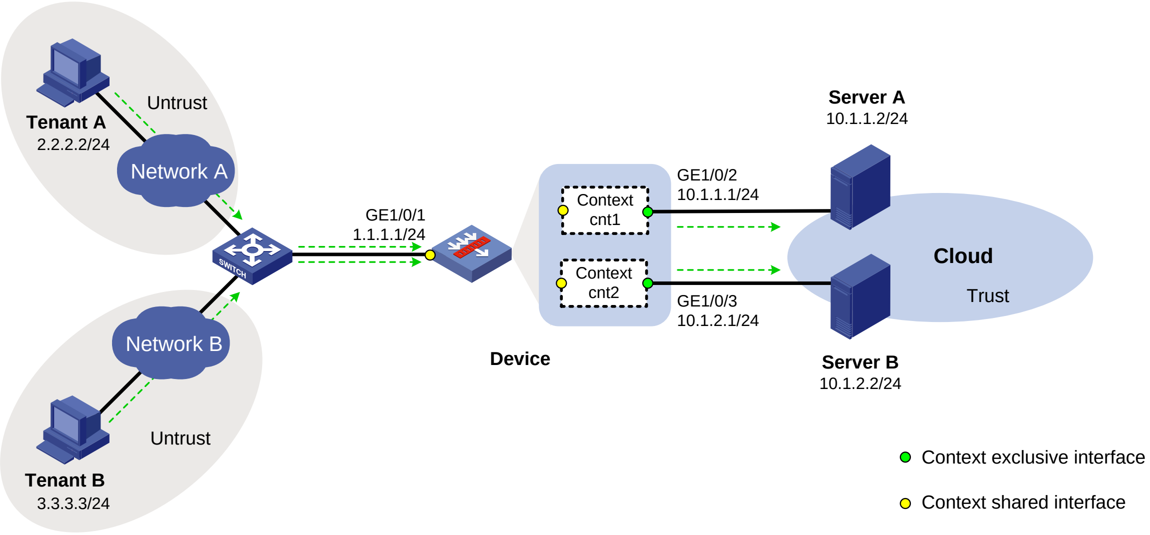
Example: Configuring contexts to act as gateways in a cloud computing center

Network configuration

As shown in Figure 5, Device is the gateway in the cloud computing center for internal information protection. The device has only one public network interface, which is GigabitEthernet 1/0/1. For Tenant A and Tenant B to independently access computing resources in the center, perform the following tasks:

- Create two contexts on the device, one is **cnt1** and the other is **cnt2**.
- Assign interfaces to the contexts for Tenant A and Tenant B to access the center.
  - Assign GigabitEthernet 1/0/1 to context **cnt1** in shared mode and GigabitEthernet 1/0/2 to the context in exclusive mode.
  - Assign GigabitEthernet 1/0/1 to context **cnt2** in shared mode and GigabitEthernet 1/0/3 to the context in exclusive mode.
- On the shared interface GigabitEthernet 1/0/1, configure a NAT internal server for Tenant A and Tenant B to use independent public IP addresses to access Server A and Server B, respectively.

Figure 5 Network diagram



Procedure

1. Assign an IP address to GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

2. Create and configure context **cnt1** for Tenant A:

# Create context **cnt1** and configure a description for the context.

```
[Device] context cnt1
[Device-context-2-cnt1] description context-1
```

# Assign GigabitEthernet 1/0/1 to context **cnt1** in shared mode.

```
[Device-context-2-cnt1] allocate interface gigabitethernet 1/0/1 share
```

# Assign GigabitEthernet 1/0/2 to context **cnt1** in exclusive mode.

```
[Device-context-2-cnt1] allocate interface gigabitethernet 1/0/2
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

# Start context **cnt1**.

```
[Device-context-2-cnt1] context start
It will take some time to start the context...
Context started successfully.
[Device-context-2-cnt1] quit
```

# Log in to context **cnt1**.

```
[Device] switchto context cnt1
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<NSFOCUS> system-view
```

# Change the system name of the context to **cnt1**.

```
[NSFOCUS] sysname cnt1
```

# Assign IP address 10.1.1.1/24 to interface GigabitEthernet 1/0/2.

```
[cnt1] interface gigabitethernet 1/0/2
[cnt1-GigabitEthernet1/0/2] ip address 10.1.1.1 24
[cnt1-GigabitEthernet1/0/2] quit
```

# Assign interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 to security zones **Untrust** and **Trust**, respectively.

```
[cnt1] security-zone name untrust
[cnt1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[cnt1-security-zone-Untrust] quit
[cnt1] security-zone name trust
[cnt1-security-zone-Trust] import interface gigabitethernet 1/0/2
[cnt1-security-zone-Trust] quit
```

# Configure the IPv4 security policy to ensure that Tenant A can access Server A.

```
[cnt1] security-policy ip
[cnt1-security-policy-ip] rule name untrust-trust
[cnt1-security-policy-ip-0-untrust-trust] action pass
[cnt1-security-policy-ip-0-untrust-trust] source-zone untrust
[cnt1-security-policy-ip-0-untrust-trust] destination-zone trust
[cnt1-security-policy-ip-0-untrust-trust] source-ip-host 2.2.2.2
[cnt1-security-policy-ip-0-untrust-trust] destination-ip-host 10.1.1.2
[cnt1-security-policy-ip-0-untrust-trust] quit
[cnt1-security-policy-ip] quit
```

# Return to the default context from context **cnt1**.

```
[cnt1] quit
<cnt1> quit
[Device]
```

3. Create and configure context **cnt2** for Tenant B:

# Create context **cnt2** and configure a description for the context.

```
[Device] context cnt2
[Device-context-3-cnt2] description context-2
```

# Assign GigabitEthernet 1/0/1 to context **cnt2** in shared mode.

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/1 share
```

# Assign GigabitEthernet 1/0/3 to context **cnt2** in exclusive mode.

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/3
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

# Start context **cnt2**.

```
[Device-context-3-cnt2] context start
It will take some time to start the context...
Context started successfully.
[Device-context-3-cnt2] quit
```

# Log in to context **cnt2**.

```
[Device] switchto context cnt2
******************************************************************************
* Copyright (c) 2004-2021 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<NSFOCUS> system-view
```

# Change the system name of the context to **cnt2**.

```
[NSFOCUS] sysname cnt2
```

# Assign IP address 10.1.2.1/24 to interface GigabitEthernet 1/0/3.

```
[cnt2] interface gigabitethernet 1/0/3
[cnt2-GigabitEthernet1/0/3] ip address 10.1.2.1 24
[cnt2-GigabitEthernet1/0/3] quit
```

# Assign interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/3 to security zones **Untrust** and **Trust**, respectively.

```
[cnt2] security-zone name untrust
[cnt2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[cnt2-security-zone-Untrust] quit
[cnt2] security-zone name trust
[cnt2-security-zone-Trust] import interface gigabitethernet 1/0/3
[cnt2-security-zone-Trust] quit
```

# Configure the IPv4 security policy to ensure that Tenant B can access Server B.

```
[cnt2] security-policy ip
[cnt2-security-policy-ip] rule name untrust-trust
[cnt2-security-policy-ip-0-untrust-trust] action pass
[cnt2-security-policy-ip-0-untrust-trust] source-zone untrust
[cnt2-security-policy-ip-0-untrust-trust] destination-zone trust
[cnt2-security-policy-ip-0-untrust-trust] source-ip-host 3.3.3.3
[cnt2-security-policy-ip-0-untrust-trust] destination-ip-host 10.1.2.2
[cnt2-security-policy-ip-0-untrust-trust] quit
[cnt2-security-policy-ip] quit
```

# Return to the default context from context **cnt2**.

```
[cnt2] quit
<cnt2> quit
[Device]
```

4. On GigabitEthernet 1/0/1, configure a NAT internal server to allow the external network to access Server A by using address http://1.1.1.2:8080 and access Server B by using address http://1.1.1.3:8080.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.2 8080 inside 10.1.1.2 http
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.3 8080 inside 10.1.2.2 http
[Device-GigabitEthernet1/0/1] quit
```

Verifying the configuration

1. Verify that the contexts are created successfully and running correctly. The device has three contexts in active state.

```
[Device] display context
ID     Name             Status       Description
1      Admin            active       DefaultContext
2      cnt1             active       context-1
3      cnt2             active       context-2
```

2. Verify that Tenant A can access Server A by using address http://1.1.1.2:8080 and Tenant B can access Server B by using address http://1.1.1.3:8080.

Contents

Configuring Reth interfaces

About Reth interfaces

A redundant Ethernet (Reth) interface is a virtual Layer 3 interface that uses two member interfaces to ensure link availability. One member interface is active and the other is inactive. When the active interface fails, the inactive interface becomes active. The member interface switchover does not interrupt traffic.

Operating mechanism

A member interface of a Reth interface can be in either of the following states:

- **Active**—The interface can forward packets. A Reth interface can have only one active member interface.
- **Inactive**—The interface cannot forward packets.

A Reth interface determines the state of its member interfaces by using the following rules:

- When the member interfaces are physically up, the member interface with the higher priority is active. The other member interface is inactive. The priority of an interface is user configurable.
- When the member interfaces are physically down, both interfaces are inactive.
- When the active member interface goes down physically, the inactive interface automatically becomes active to forward packets.

If the Reth interface is added to a redundancy group, the member interface states are determined by the redundancy group. For more information, see "Configuring redundancy groups."

The switchover between the Reth member interfaces is not visible to the network and does not cause network topology changes. For the upstream and downstream devices, they are connected to the Reth interface and learn only the MAC address of the Reth interface.

Application scenario

The Reth interface feature is typically used with the redundancy group feature. For more information, see "Configuring redundancy groups."

Reth subinterfaces

To transmit and receive VLAN-tagged packets on a Reth interface, you can create subinterfaces for the Reth interface. A Reth subinterface is a Layer 3 logical interface and can be assigned an IP address. Packets from different VLANs can be forwarded by different Reth subinterfaces, which improves the interface efficiency and networking flexibility. For more information about VLAN-tagged packet processing on a Reth subinterface, see VLAN termination in *Layer 2—LAN Switching Configuration Guide*.

Restrictions: Hardware compatibility with Reth interface

Models	Reth interface compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: Reth interface configuration

During master/subordinate switchover in an IRF fabric, do not delete the aggregate interface from the Reth interface.

Configuring a Reth interface

Restrictions and guidelines: Reth interface configuration

Supported member interface types

A Reth interface can use the following interfaces and their subinterfaces as member interfaces:

- Layer 3 Ethernet interfaces.

IMPORTANT:

If a Layer 3 Ethernet interface is marked as a bypass interface on the panel or has the bypass feature enabled, do not use that interface as a member interface of a Reth interface. If you do so, communication errors will occur. For more information about the bypass feature, see "Configuring bridge forwarding."

- Layer 3 aggregate interfaces.

Member interface configuration

When you configure a Reth interface, follow these restrictions and guidelines:

- You can assign a maximum of two member interfaces to a Reth interface. The member interfaces must have different priorities.
- You cannot assign member interfaces to a Reth subinterface.
- You cannot assign subinterfaces or interfaces that have subinterfaces to a Reth interface if the Reth interface has Reth subinterfaces.
- You can assign an interface or subinterface to only one Reth interface.
- As a best practice, assign interfaces of the same type and speed to a Reth interface.
- If both member interfaces of a Reth interface are subinterfaces, make sure they are on different main interfaces and terminate the same VLAN ID. For more information about VLAN termination, see *Layer 2—LAN Switching Configuration Guide*.

- Do not specify a Reth interface as the outgoing interface in IPv6 static neighbor entries if its member interfaces contain subinterfaces. For more information about IPv6 static neighbor entries, see *Layer 3—IP Services Configuration Guide*.
- Settings made on the member interfaces of a Reth interface will not take effect until they are removed from the Reth interface.

Member interface deletion

Before you delete a Reth interface, make sure all its member interfaces have been removed.

Configuring basic parameters for a Reth interface

1. Enter system view.

   `system-view`

2. Create a Reth interface and enter its view.

   `interface reth` *`interface-number`*

   By default, no Reth interfaces exist.

3. Assign a member interface to the Reth interface.

   `member interface` *`interface-type interface-number`* **`priority`** *`priority`*

   By default, a Reth interface does not have member interfaces.

4. (Optional.) Configure the expected bandwidth for the Reth interface.

   `bandwidth` *`bandwidth-value`*

   By default, the expected bandwidth is 10000 kbps.

   The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

5. (Optional.) Configure the description of the Reth interface.

   `description` *`text`*

   The default description of a Reth interface is *interface-name* **Interface** (for example, **Reth1 Interface**).

6. (Optional.) Set the MTU of the Reth interface.

   `mtu` *`size`*

   By default, the MTU of a Reth interface is 1500 bytes.

7. (Optional.) Assign a MAC address to the Reth interface.

   `mac-address` *`mac-address`*

   By default, Reth interfaces use the bridge MAC address of the device.

   Reth interfaces use the same default MAC address. To avoid communication errors, manually assign unique MAC addresses to the Reth interfaces.

8. Bring up the Reth interface.

   `undo shutdown`

   By default, a Reth interface is not manually shut down.

Setting the parameters for retransmitting advertisement messages after a Reth member interface switchover

About this task

After you configure the parameters for retransmitting advertisement messages, a Reth interface performs the following operations when a Reth member interface switchover occurs on it:

1. Sends advertisement messages (including gratuitous ARP messages and NA messages) to neighbors immediately.
2. Retransmits the advertisement messages according to the number of retransmissions and the retransmission interval you have configured.

Restrictions and guidelines

If a Reth interface has subinterfaces, the subinterfaces also send advertisement messages upon a Reth member interface switchover. To save CPU resources, the parameters for retransmitting advertisement messages take effect only on Reth interfaces. Reth subinterfaces are not controlled by these parameters.

Procedure

1. Enter system view.
   `system-view`
2. Set the parameters for retransmitting advertisement messages to neighbors after a Reth member interface switchover.
   `reth advertise retransmit` *`times`* **`interval`** *`seconds`*
   By default, after a Reth member interface switchover, a Reth interface retransmits advertisement messages to neighbors five times at an interval of 1 second.

Enabling fast traffic switchover on a Reth interface

About fast traffic switchover

IMPORTANT:

This feature introduces low possibility of forwarding traffic on the inactive member interface while the system is operating correctly. Make sure you understand this impact on your services when you use this feature.

This feature enables faster traffic switchover between Reth member interfaces than the standard switchover mechanism of Reth when the master device is powered off or reboots unexpectedly.

This feature implements fast switchover by allowing the inactive member interface to forward packets. In rare cases, the neighbor device might learn MAC address entries on the link connected to the inactive interface and sends traffic to the inactive interface.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080	Yes
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

For this feature to take effect and operate effectively, follow these restrictions and guidelines:

- Make sure the Reth member interfaces are physical interfaces.
- Enable fast switchover on both the Reth interface for uplink traffic and the Reth interface for downlink traffic.
- Assign the downlink and uplink Reth interfaces to a redundancy group.

- Make sure the high-priority Reth member interfaces are on the master device (high-priority redundancy group node).
- To minimize the chance of receiving traffic on the inactive interface, use the **`arp timer aging`** command on the neighbor device to shorten the ARP entry aging timer.

Procedure

1. Enter system view.

   `system-view`

2. Enter Reth interface view.

   `interface reth` *`interface-number`*

3. Enable fast traffic switchover.

   `fast-switch enable`

   By default, fast traffic switchover is disabled.

Configuring a Reth subinterface

About this task

You can create Reth subinterfaces and configure VLAN termination on the Reth subinterfaces to process VLAN-tagged packets.

Restrictions and guidelines

When you configure a Reth subinterface, follow these restrictions and guidelines:

- To create a Reth subinterface, create the Reth interface first.
- You cannot create subinterfaces for a Reth interface in any of the following situations:
  - The members of the Reth interface are Layer 3 Ethernet subinterfaces or Layer 3 aggregate subinterfaces.
  - A minimum of one subinterface is created on the member interfaces of the Reth interface.

Procedure

1. Enter system view.

   `system-view`

2. Create a Reth subinterface and enter its view.

   `interface reth` *`interface-number.subnumber`*

   By default, no Reth subinterfaces exist.

   If the specified Reth subinterface already exists, this command enters the view of the Reth subinterface.

3. (Optional.) Configure the expected bandwidth for the Reth subinterface.

   `bandwidth` *`bandwidth-value`*

   By default, the expected bandwidth is 10000 kbps.

4. (Optional.) Configure a description for the Reth subinterface.

   `description` *`text`*

   The default description of a Reth subinterface is *interface-name* **Interface** (for example, **Reth1.1 Interface**).

5. (Optional.) Set the MTU of the Reth subinterface.

   `mtu` *`size`*

   By default, the MTU of a Reth subinterface is 1500 bytes.

6. Return to system view.

quit

7. Enter Reth interface view.
   interface reth *interface-number*
8. Enable subinterface rate statistics collection on the Reth interface.
   sub-interface rate-statistic

   By default, subinterface rate statistics collection is disabled on a Reth interface.

   After you execute this command, the device periodically refreshes subinterface rate statistics for the Reth interface. The statistics is displayed in the **Last 300 seconds input rate** and **Last 300 seconds output rate** fields of the command output from the **display interface reth** command.
9. Return to system view.
   quit
10. Enter Reth subinterface view.
    interface reth *interface-number.subnumber*
11. Bring up the Reth subinterface.
    undo shutdown

    By default, a Reth subinterface is not manually shut down.

Restoring default settings for a Reth interface or subinterface

Restrictions and guidelines

CAUTION:

The **default** command might interrupt ongoing network services. Make sure you are fully aware of the impacts of this command when you execute it on a live network.

This command might fail to restore the default settings for some commands for reasons such as command dependencies and system restrictions.

To resolve this issue:

1. Use the **display this** command in interface view to identify these commands.
2. Use their **undo** forms or follow the command reference to restore their default settings.
3. If the restoration attempt still fails, follow the error message instructions to resolve the issue.

Procedure

1. Enter system view.
   system-view
2. Enter a Reth interface or subinterface view.
   interface reth { *interface-number* | *interface-number.subnumber* }
3. Restore the default settings for the Reth interface.
   default

Displaying and maintaining Reth interfaces

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display Reth interface traffic statistics.	**display counters** { **inbound** | **outbound** } **interface** [**reth** [*interface-number*]]
Display traffic rate statistics for Reth interfaces in up state during the most recent statistics polling interval.	**display counters rate** { **inbound** | **outbound** } **interface** [**reth** [*interface-number*]]
Display Reth interface or subinterface information.	**display interface** [**reth** [*interface-number* | *interface-number.subnumber*]] [**brief** [**description** | **down**]]
Display information about the member interfaces of a Reth interface.	**display reth interface** *interface-type interface-number*
Clear statistics for Reth interfaces.	**reset counters interface** [**reth** [*interface-number*]]

Configuring redundancy groups

About redundancy groups

A redundancy group works on IRF fabrics. It allows traffic to enter and leave an IRF fabric through the same member device.

Operating mechanism of a redundancy group

As shown in Figure 1, a redundancy group has two redundancy group nodes. Each node is bound to an IRF member device.

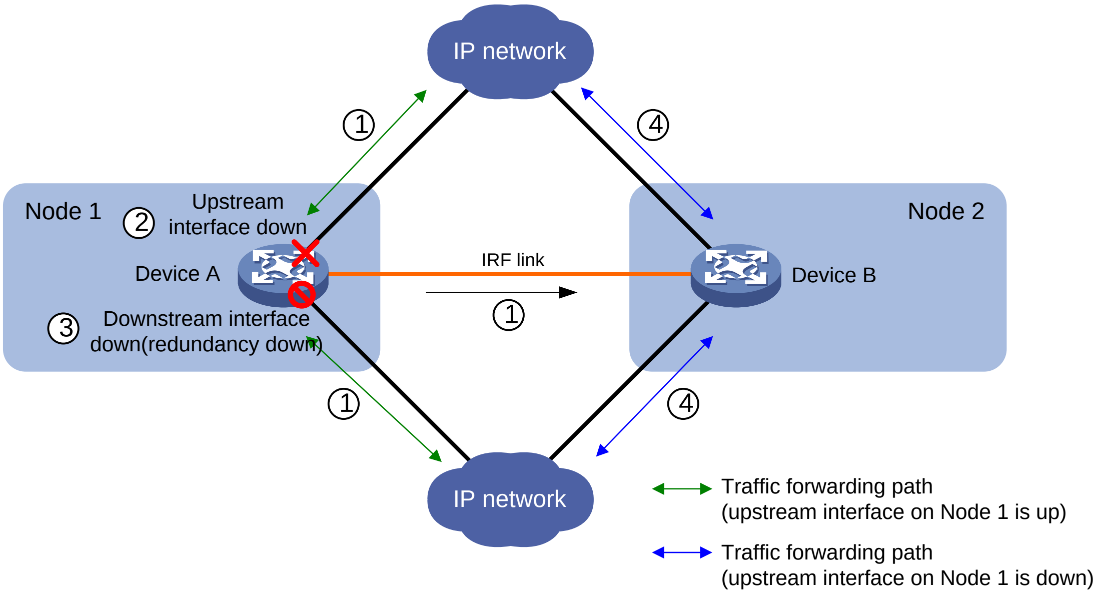
A redundancy group node is a collection of objects on its bound IRF member device. The objects include member interfaces of Reth interfaces and individual physical Ethernet interfaces. The state of the objects is consistent with the state of the node.

In a redundancy group, one node is in primary state, and the other node is in secondary state. Only the primary node forwards traffic. When the primary node fails, the redundancy group switches over to the secondary node. This mechanism ensures path symmetry for traffic.

As shown in Figure 1, a redundancy group performs a switchover as follows:

1. When both IRF member devices are operating correctly, the redundancy group forwards traffic through Node 1 (Device A) and backs up services and data (such as NAT) to Node 2 (Device B).
2. When the upstream interface on Device A fails, the redundancy group shuts down the downstream interface on Device A and switches traffic over to Device B.

Figure 1 Redundancy group operating mechanism


IP network
①
④
Node 1
② Upstream interface down
Node 2
Device A
IRF link
Device B
①
③ Downstream interface down(redundancy down)
①
④
IP network
Traffic forwarding path (upstream interface on Node 1 is up)
Traffic forwarding path (upstream interface on Node 1 is down)

Redundancy group node states

A redundancy group node can act as the primary or secondary node. Only the primary node can forward traffic.

When both nodes are operating correctly, the primary node is selected in the following order:

1. The node with higher node priority.
2. The node with smaller ID if the two nodes have the same priority.

When the primary node fails, the secondary node takes over the primary role to forward traffic. For more information about the state monitoring and switchover mechanisms, see "Redundancy group switchover."

Redundancy group members

Application scenarios of redundancy group members

Redundancy group members can be physical Ethernet interfaces and Reth interfaces that are located on the IRF member devices bound to the group's nodes.

You can assign physical Ethernet interfaces or Reth interfaces to a redundancy group for symmetric traffic forwarding, as shown in Table 1.

Table 1 Application scenarios of physical Ethernet interfaces and Reth interfaces

Member type	Application scenarios	Supported interfaces
Physical Ethernet interfaces	Dynamic routing protocols run between the IRF fabric and its upstream and downstream devices.	Layer 2 Ethernet interfaces. Later 3 Ethernet interfaces.
Reth interfaces	No dynamic routing protocol runs between the IRF fabric and its upstream and downstream devices.	A Reth interface can use the following interfaces and their subinterfaces as member interfaces: • Layer 3 Ethernet interfaces. • Layer 3 aggregate interfaces.

Using physical Ethernet interfaces in a redundancy group

You assign physical Ethernet interfaces to a redundancy group by binding them to their respective redundancy group nodes.

For symmetric traffic switchover, you must bind a minimum of one downlink interface and a minimum of one uplink interface with each node of the redundancy group.

The state of the member physical Ethernet interfaces changes with the state of the redundancy group nodes. Only the member interfaces on the primary node can forward traffic.

As shown in Figure 2, Interface A1 and Interface A2 are on Node 1, and Interface B1 and Interface B2 are on Node 2. When Node 1 is in primary state, Interface A1 and Interface A2 are up to forward traffic, while Interface B1 and Interface B2 do not forward traffic.

When Interface A1 goes down, the Reth module places Node 1 in secondary state. Node 2 changes to the primary state, and Interface B1 and Interface B2 take over to forward traffic, as shown in Figure 3.

Figure 2 States of the member interfaces when both nodes are operating correctly

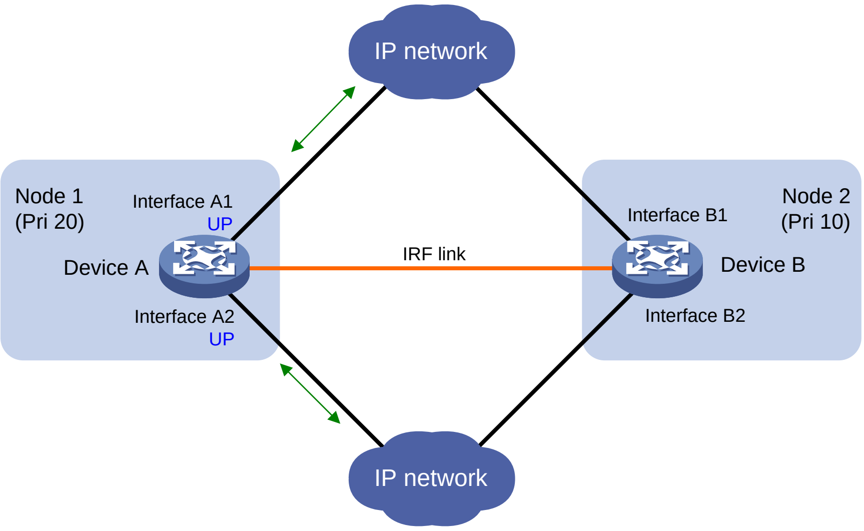
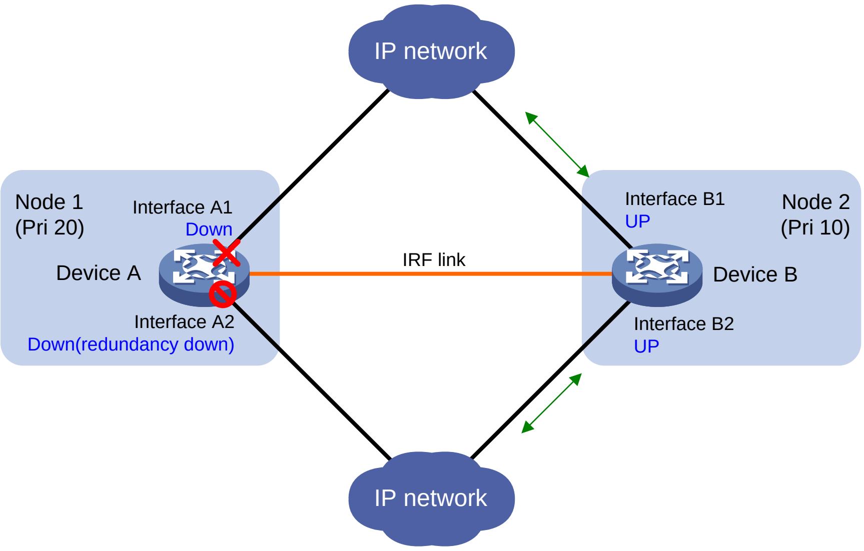


Figure 3 States of the member interfaces after a switchover



Using Reth interfaces in a redundancy group

To use Reth interfaces for symmetric forwarding, you must assign two Reth interfaces to a redundancy group: one for uplink traffic and the other for downlink traffic. The Reth interfaces must meet the following requirements:

- The Reth interface for uplink traffic contains one uplink port on each redundancy group node.
- The Reth interface for downlink traffic contains one downlink port on each redundancy group node.
- The high-priority member of each Reth interface belongs to the high-priority node.

The state of each Reth interface's members depends on the state of the redundancy group nodes.

- When the high-priority node is in primary state, the high-priority member is active.
- When the low-priority node is in primary state, the low-priority member is active.

As shown in Figure 4, redundancy group 1 contains Reth 1 for uplink traffic and Reth 2 for downlink traffic. Reth 1 contains Interface A1 (on Node 1) and Interface B1 (on Node 2). Reth 2 contains Interface A2 (on Node 1) and Interface B2 (on Node 2).

When Node 1 is in primary state, Interface A1 in Reth 1 and Interface A2 in Reth 2 are active to forward uplink and downlink traffic, respectively.

When Interface A1 fails, the Reth module places Node 1 in secondary state and shuts down Interface A2, as shown in Figure 5. Node 2 changes to the primary state, and Interface B1 and Interface B2 become active to forward uplink and downlink traffic.

Figure 4 States of each Reth interface's members when both nodes are operating correctly

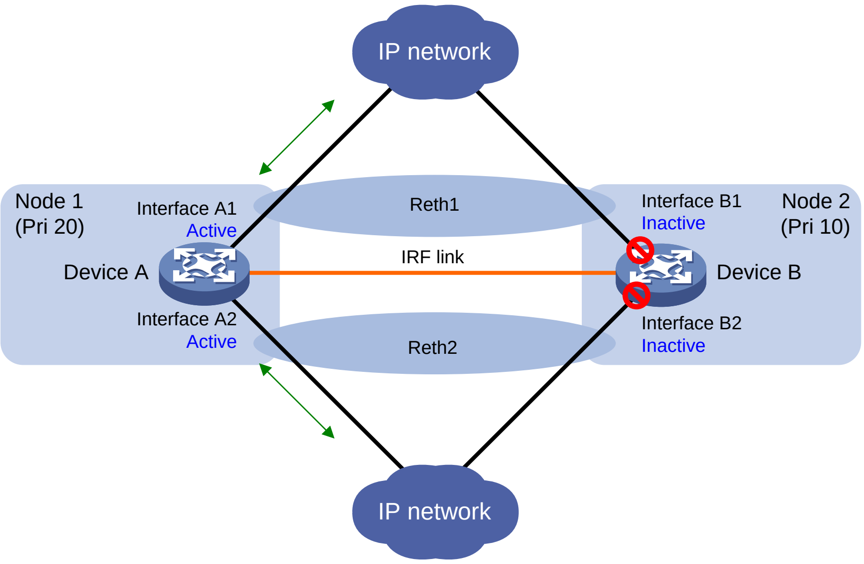
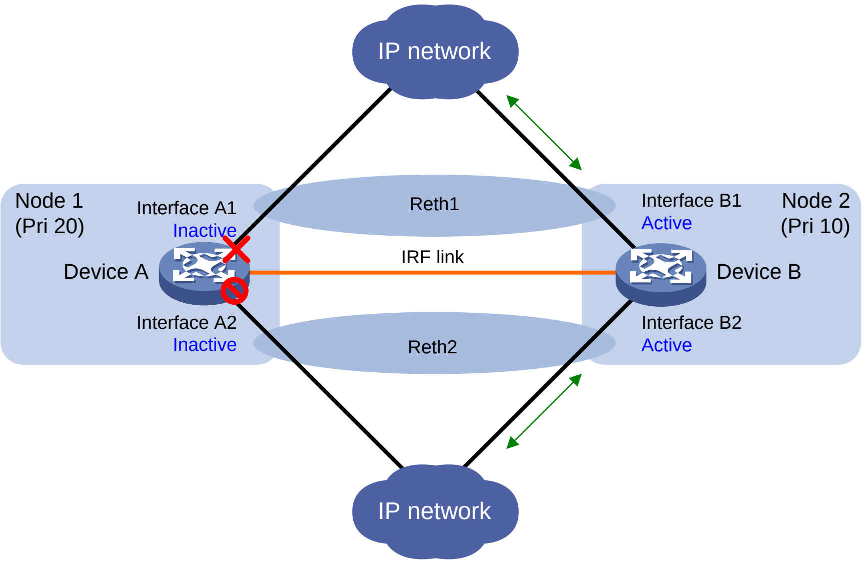


Figure 5 States of each Reth interface's members after a switchover



Redundancy group switchover

Switchover types

Redundancy group switchovers include automatic switchovers and manual switchovers.

Automatic switchover timers

Timers for automatic switchovers include the hold-down timer and the preemption delay timer.

- **Hold-down timer**—The hold-down timer specifies the minimum interval between two switchovers to prevent frequent switchovers. The timer starts when a switchover is finished. The redundancy group can perform the next switchover only after the hold-down timer expires.
- **Preemption delay timer**—The preemption delay timer specifies the delay for a switchover back to the high-priority node. The preemption delay timer starts when the switchover is triggered. The redundancy group performs the switchover only after the timer expires. The delay allows the system to process events (such as interface state changes) required for the switchover.

Automatic switchover

A redundancy group cooperates with the Track module to monitor link and interface status for automatic switchovers.

A redundancy group node has a weight of 255 (not configurable). Each redundancy group node is associated with one or multiple track entries that have a user-configurable weight decrement rate. When the state of a track entry changes, the weight of the associated node is reduced or increased, as follows:

- When the track entry changes to the **NotReady** or **Negative** state, the node weight is reduced by the weight decrement rate of the track entry.
- When the track entry changes to the **Positive** state, the node weight is increased by the weight decrement rate of the track entry.

When the node weight decreases to 0 or a lower value, a switchover request is triggered.

- Switchover to the low-priority node occurs when the hold-down timer expires.
- Switchover to the high-priority node occurs when the preemption delay timer expires.

Manual switchover

You can issue a manual switchover request in one of the following situations:

- An automatic switchover to the high-priority node cannot be performed when no tracked interfaces are excluded from the shutdown action by the Reth module.
- Switchovers are required when both redundancy group nodes operate correctly. For example, component replacement is required on the high-priority node.

Automatic switchover to the high-priority node is not allowed if the preemption delay timer is set to 0, even when both nodes are operating correctly. You can perform only manual switchover.

Restrictions: Hardware compatibility with redundancy group

Models	Redundancy group compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Redundancy group configuration tasks at a glance

To configure a redundancy group, perform the following tasks:

1. Creating a redundancy group

2. Configuring a redundancy group node
3. Assigning members to the redundancy group
   - Assigning physical Ethernet interfaces to a redundancy group

     This task is applicable to the network scenario where a dynamic routing protocol runs between the IRF fabric and its upstream and downstream devices.
   - Assigning Reth interfaces to a redundancy group

     This task is applicable to the network scenario where no dynamic routing protocol runs between the IRF fabric and its upstream and downstream devices.
4. Configuring the switchover timers
5. (Optional.) Performing a manual switchover
6. (Optional.) Performing a manual switchover back to the high-priority node
7. (Optional.) Enabling SNMP notifications for redundancy groups

Creating a redundancy group

Restrictions and guidelines

Before you delete a redundancy group, you must remove all its Reth interfaces and redundancy group nodes.

Procedure

1. Enter system view.

   `system-view`
2. Create a redundancy group and enter its view.

   `redundancy group` *`group-name`*

   By default, no redundancy groups exist.

Configuring a redundancy group node

Restrictions and guidelines

You can configure a maximum of two nodes for a redundancy group. Nodes in different redundancy groups can use the same ID.

You can bind a redundancy group node to one IRF member device. An IRF member device cannot be bound to multiple redundancy group nodes. You cannot change the binding for a node if it has member interfaces or is associated with track entries.

When you associate a track entry with a redundancy group node, follow these restrictions and guidelines:

- You cannot associate a track entry with both nodes in a redundancy group.
- For correct interface state recovery, you must exclude a tracked interface from the shutdown action by the Reth module if the interface has one of the following roles:
  - Member interface in the redundancy group.
  - Member of a Reth interface in the redundancy group.
- On the high-priority node, do not exclude a subinterface from the shutdown action by the Reth module if its main interface has one of the following roles:
  - Member interface of the redundancy group.
  - Member of a Reth interface in the redundancy group.

When the Reth module shuts down the main interface, the subinterface is also shut down. The shutdown subinterface cannot recover automatically to trigger an automatic switchover.

Prerequisites

Before you associate a track entry with a redundancy group, you must configure the track entry. For more information about configuring track entries, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter redundancy group view.

   redundancy group *group-name*

3. Create a redundancy group node and enter its view.

   node *node-id*

   By default, no redundancy group nodes exist.

4. Set the priority of the redundancy group node.

   priority *priority*

   By default, the priority of a redundancy group node is 1.

5. Bind the redundancy group node with an IRF member device.

   bind slot *slot-number*

   By default, a node is not bound with an IRF member device.

6. Associate an existing track entry with the redundancy group node.

   track *track-entry-number* [**reduced** *weight-reduced*] [**interface** *interface-type interface-number*]

   By default, a node is not associated with track entries.

Assigning physical Ethernet interfaces to a redundancy group

Restrictions and guidelines

For symmetric traffic switchover, you must bind a minimum of one uplink interface and a minimum of one downlink interface with each node of the redundancy group.

You can bind a physical Ethernet interface with only one redundancy group node.

You cannot bind a member of a Reth interface with a redundancy group node.

Procedure

1. Enter system view.

   system-view

2. Enter redundancy group view.

   redundancy group *group-name*

3. Enter redundancy group node view.

   node *node-id*

4. Bind a physical Ethernet interface with the redundancy group node.

   node-member interface *interface-type interface-number*

   By default, a physical Ethernet interface is not bound with a redundancy group node.

Assigning Reth interfaces to a redundancy group

Restrictions and guidelines

To use Reth interfaces for symmetric forwarding, you must assign two Reth interfaces to a redundancy group: one for uplink traffic and the other for downlink traffic. The Reth interfaces must meet the following requirements:

- The Reth interface for uplink traffic contains one uplink port on each redundancy group node.
- The Reth interface for downlink traffic contains one downlink port on each redundancy group node.
- The high-priority member of each Reth interface belongs to the high-priority node.

Procedure

1. Enter system view.

   `system-view`

2. Create a Reth interface and enter its view.

   `interface reth` *`interface-number`*

3. Assign a member interface to the Reth interface.

   `member interface` *`interface-type interface-number`* **`priority`** *`priority`*

   By default, a Reth interface does not have member interfaces.

   Repeat this step to assign two member interfaces to the Reth interface. Assign a higher priority for the *`priority`* argument to the member interface on the high-priority redundancy node.

4. Return to system view.

   `quit`

5. Enter redundancy group view.

   `redundancy group` *`group-name`*

6. Assign the Reth interface to the redundancy group.

   `member interface reth` *`interface-number`* `[` **`quick-fallback`** `]`

   By default, a redundancy group does not contain Reth interfaces.

Configuring the switchover timers

1. Enter system view.

   `system-view`

2. Enter redundancy group view.

   `redundancy group` *`group-name`*

3. Set the hold-down timer for the redundancy group.

   `hold-down-interval` *`second`*

   By default, the hold-down timer is 1 second.

4. Set the preemption delay timer for the redundancy group.

   `preempt-delay seconds` *`sec`*

   By default, the preemption delay timer is 1 minute (60 seconds).

   If you set this timer to 0 seconds, automatic switchover to the high-priority node is disabled. You can perform only manual switchover.

Performing a manual switchover to the low-priority node

1. Enter system view.
   `system-view`
2. Enter redundancy group view.
   `redundancy group` *`group-name`*
3. Request a switchover to the low-priority node.
   `switchover request`

Performing a manual switchover back to the high-priority node

1. Enter system view.
   `system-view`
2. Enter redundancy group view.
   `redundancy group` *`group-name`*
3. Request a switchover back to the high-priority node.
   `switchover reset`

Enabling SNMP notifications for redundancy groups

About this task

This feature enables SNMP notifications for the following events:

- A manual switchover is performed.
- An interface goes down.
- A faulty interface is recovered.

To send the event notifications to an NMS, you must also configure SNMP as described in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.
   `system-view`
2. Enable SNMP notifications for redundancy groups.
   `snmp-agent trap enable rddc`
   By default, SNMP notifications are enabled for redundancy groups.

Displaying and maintaining redundancy groups

Execute **`display`** commands in any view.

Task	Command
Display redundancy group information.	**display redundancy group** [*group-name*]

Redundancy group configuration examples

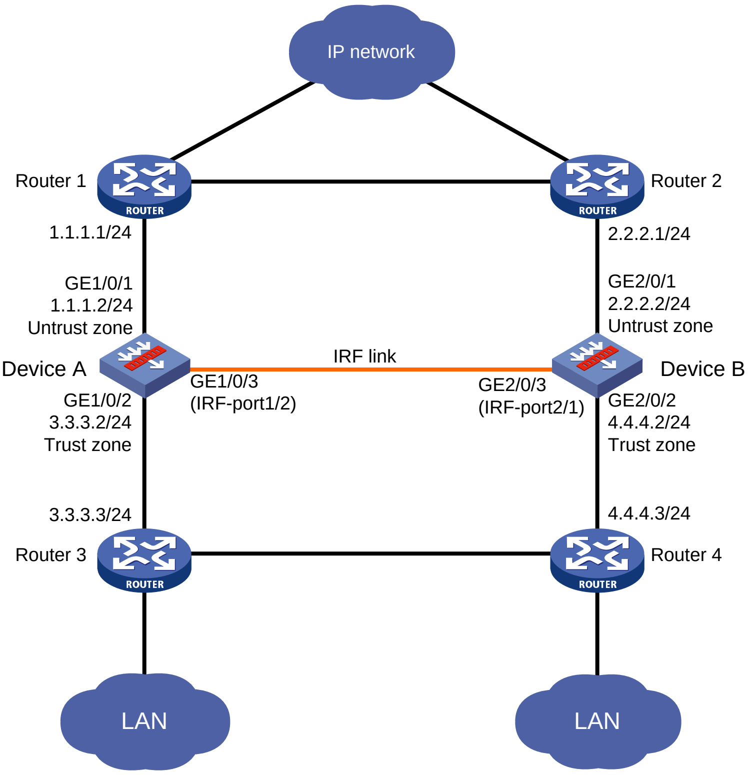
Example: Configuring a redundancy group with Layer 3 interface members

Network configuration

As shown in Figure 6, Device A (member ID 1) and Device B (member ID 2) form an IRF fabric.

Configure a redundancy group on the IRF fabric to ensure that traffic is forwarded along the Router 1—Device A—Router 3 path when the path is available. When a link or device failure occurs on the path, switch traffic to the Router 2—Device B—Router 4 path.

Figure 6 Network diagram



Procedure

1. Configure IRF:
   a. Configure Device A:

   # Bind GigabitEthernet 1/0/3 to IRF port 1/2, and save the configuration.

```
<DeviceA> system-view
[DeviceA] interface GigabitEthernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] shutdown
[DeviceA-GigabitEthernet1/0/3] quit
```

```
[DeviceA] irf-port 1/2
[DeviceA-irf-port1/2] port group interface GigabitEthernet 1/0/3
You must perform the following tasks for a successful IRF setup:
Save the configuration after completing IRF configuration.
Execute the "irf-port-configuration active" command to activate the IRF ports.
[DeviceA-irf-port1/2] quit
[DeviceA] interface GigabitEthernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] undo shutdown
[DeviceA-GigabitEthernet1/0/3] quit
[DeviceA] save
The current configuration will be written to the device. Are you sure? [Y/N]:y
Please input the file name(*.cfg)[flash:/startup.cfg]
(To leave the existing filename unchanged, press the enter key):irf-port-configu
ration active
The configuration file is invalid or not exist.
```

# Assign member priority 2 to Device A for it to becomes the IRF master, and activate the IRF port configuration.

```
[DeviceA] irf member 1 priority 2
[DeviceA] irf-port-configuration active
```

b. Configure Device B:

# Change the member ID of Device B to 2, and reboot the device to validate the change.

```
<DeviceB> system-view
[DeviceB] irf member 1 renumber 2
Renumbering the member ID may result in configuration change or loss. Continue?
[Y/N]:y
[DeviceB] quit
<DeviceB> reboot
Start to check configuration with next startup configuration file, please wait..
.......DONE!
This command will reboot the device. Continue? [Y/N]:y
```

# Connect Device B to Device A (see Figure 6).

# Log in to Device B. (Details not shown.)

# Bind GigabitEthernet 2/0/3 to IRF port 2/1, save the configuration, and activate the IRF port configuration.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 2/0/3
[DeviceB-GigabitEthernet2/0/3] shutdown
[DeviceB-GigabitEthernet2/0/3] quit
[DeviceB] irf-port 2/1
[DeviceB-irf-port2/1] port group interface GigabitEthernet 2/0/3
You must perform the following tasks for a successful IRF setup:
Save the configuration after completing IRF configuration.
Execute the "irf-port-configuration active" command to activate the IRF ports.
[DeviceB-irf-port2/1] quit
[DeviceB] interface GigabitEthernet 2/0/3
[DeviceB-GigabitEthernet2/0/3] undo shutdown
[DeviceB-GigabitEthernet2/0/3] quit
[DeviceB] save
```

```
[DeviceB] irf-port-configuration active
```

2. Create track entries to monitor the physical layer state of the interfaces.

```
<DeviceA> system-view
[DeviceA] track 1 interface gigabitethernet 1/0/1 physical
[DeviceA-track-1] quit
[DeviceA] track 2 interface gigabitethernet 1/0/2 physical
[DeviceA-track-2] quit
[DeviceA] track 3 interface gigabitethernet 2/0/1 physical
[DeviceA-track-3] quit
[DeviceA] track 4 interface gigabitethernet 2/0/2 physical
[DeviceA-track-4] quit
```

3. Configure a redundancy group:

# Create redundancy group **aaa** and create node 1 for the redundancy group.

```
[DeviceA] redundancy group aaa
[DeviceA] redundancy-group-aaa] node 1
```

# Bind node 1 to Device A.

```
[DeviceA-redundancy-group-aaa-node1] bind slot 1
```

# Set the priority of node 1 to 100.

```
[DeviceA-redundancy-group-aaa-node1] priority 100
```

# Assign GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 to node 1.

```
[DeviceA-redundancy-group-aaa-node1] node-member interface gigabitethernet 1/0/1
[DeviceA-redundancy-group-aaa-node1] node-member interface gigabitethernet 1/0/2
```

# Associate track entries 1 and 2 with node 1. Exclude GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 from the shutdown action by the Reth module.

```
[DeviceA-redundancy-group-aaa-node1] track 1 interface gigabitethernet 1/0/1
[DeviceA-redundancy-group-aaa-node1] track 2 interface gigabitethernet 1/0/2
[DeviceA-redundancy-group-aaa-node1] quit
```

# Create node 2 for redundancy group **aaa**.

```
[DeviceA] redundancy-group-aaa] node 2
```

# Bind node 2 to Device B.

```
[DeviceA-redundancy-group-aaa-node2] bind slot 2
```

# Set the priority of node 2 to 50.

```
[DeviceA-redundancy-group-aaa-node2] priority 50
```

# Assign GigabitEthernet 2/0/1 and GigabitEthernet 2/0/2 to node 2.

```
[DeviceA-redundancy-group-aaa-node2] node-member interface gigabitethernet 2/0/1
[DeviceA-redundancy-group-aaa-node2] node-member interface gigabitethernet 2/0/2
```

# Associate track entries 3 and 4 with node 2. Exclude GigabitEthernet 2/0/1 and GigabitEthernet 2/0/2 from the shutdown action by the Reth module.

```
[DeviceA-redundancy-group-aaa-node2] track 3 interface gigabitethernet 2/0/1
[DeviceA-redundancy-group-aaa-node2] track 4 interface gigabitethernet 2/0/2
[DeviceA-redundancy-group-aaa-node2] quit
[DeviceA-redundancy-group-aaa] quit
```

4. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

5. Configure settings for routing.

   This example configures static routes, and the next hops in the routes are 1.1.1.1, 2.2.2.1, 3.3.3.3, and 4.4.4.3.

```
[DeviceA] ip route-static 0.0.0.0 0 1.1.1.1
[DeviceA] ip route-static 0.0.0.0 0 2.2.2.1 preference 80
[DeviceA] ip route-static 5.5.5.0 24 3.3.3.3
[DeviceA] ip route-static 5.5.5.0 24 4.4.4.3 preference 80
```

6. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 2/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] import interface gigabitethernet 2/0/2
[DeviceA-security-zone-Trust] quit
```

7. Configure a security policy. Configure a rule named **trust-untrust** to permit the packets between the LANs and the external network.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 5.5.5.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

Verifying the configuration

# Verify that node 1 is the primary node in redundancy group **aaa**. The member interfaces are up on both node 1 and node 2.

```
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID      Slot       Priority   Status        Track weight
  1            Slot1      100        Primary       255
  2            Slot2      50         Secondary     255

Preempt delay time remained     : 0    min
Preempt delay timer setting     : 1    min
Remaining hold-down time        : 0    sec
Hold-down timer setting         : 1    sec
Manual switchover request       : No

Node 1:
  Node member     Physical status
      GE1/0/1   UP
      GE1/0/2   UP
  Track info:
    Track    Status       Reduced weight     Interface
    1        Positive     255                GE1/0/1
    2        Positive     255                GE1/0/2
Node 2:
  Node member     Physical status
      GE2/0/1   UP
      GE2/0/2   UP
  Track info:
    Track    Status       Reduced weight     Interface
    3        Positive       255              GE2/0/1
    4        Positive       255              GE2/0/2
```

# Shut down GigabitEthernet 1/0/2.

```
[DeviceA] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] shutdown
```

# Verify that node 2 has become the primary node in redundancy group **aaa**. GigabitEthernet 1/0/1 has been shut down by the Reth module. The member interfaces on node 2 are up.

```
[DeviceA-GigabitEthernet1/0/2] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID      Slot       Priority   Status        Track weight
  1            Slot1      100        Secondary     -255
  2            Slot2      50         Primary       255

Preempt delay time remained     : 0    min
Preempt delay timer setting     : 1    min
Remaining hold-down time        : 0    sec
Hold-down timer setting         : 1    sec
Manual switchover request       : No

Node 1:
  Node member     Physical status
      GE1/0/1       DOWN(redundancy down)
      GE1/0/2       DOWN
  Track info:
    Track    Status       Reduced weight     Interface
    1         Negative     255                GE1/0/1
    2         Negative     255                GE1/0/2 (Fault)
Node 2:
  Node member     Physical status
      GE2/0/1   UP
      GE2/0/2   UP
  Track info:
    Track    Status       Reduced weight     Interface
    3         Positive       255              GE2/0/1
    4         Positive       255              GE2/0/2
```

Example: Configuring a redundancy group with Reth interface members

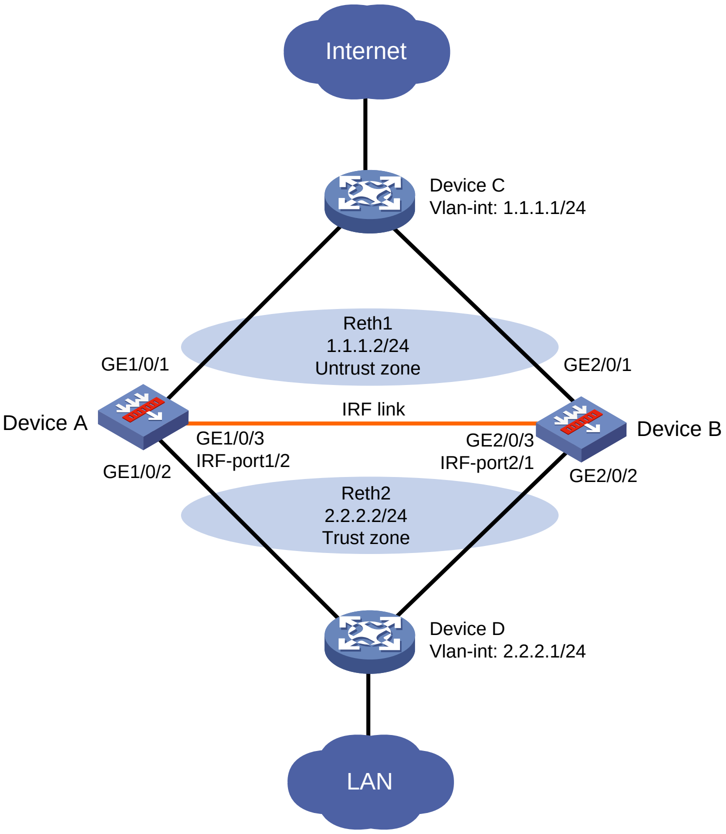
Network configuration

As shown in Figure 7, Device A (member ID 1) and Device B (member ID 2) form an IRF fabric. The IRF fabric connects to Device C and Device D through Reth 1 and Reth 2, respectively. Device C and Device D each use a VLAN interface to communicate with the IRF fabric.

Configure a redundancy group on the IRF fabric to ensure that traffic is forwarded along the Device C—Device A—Device D path when the path is available. When a link or device failure occurs on the path, traffic is switched to the Device C—Device B—Device D path.

This configuration example provides only redundancy group configuration. For more information about IRF fabric setup, see *Virtual Technologies Configuration Guide*.

Figure 7 Network diagram



Procedure

1. Configure IRF as described in "Example: Configuring a redundancy group with Layer 3 interface members."
2. Configure Reth interfaces:

   # Create Reth 1 and enter its view.

   ```
   <DeviceA> system-view
   [DeviceA] interface reth 1
   ```

   # Assign an IP address to Reth 1.

   ```
   [DeviceA-Reth1] ip address 1.1.1.2 24
   ```

# Assign GigabitEthernet 1/0/1 and GigabitEthernet 2/0/1 to Reth 1, and set their priority to 255 and 50, respectively.

```
[DeviceA-Reth1] member interface gigabitethernet 1/0/1 priority 255
[DeviceA-Reth1] member interface gigabitethernet 2/0/1 priority 50
[DeviceA-Reth1] quit
```

# Create Reth 2 and enter its view.

```
[DeviceA] interface reth 2
```

# Assign an IP address to Reth 2.

```
[DeviceA-Reth2] ip address 2.2.2.2 24
```

# Assign GigabitEthernet 1/0/2 and GigabitEthernet 2/0/2 to Reth 2, and set their priority to 255 and 50, respectively.

```
[DeviceA-Reth2] member interface gigabitethernet 1/0/2 priority 255
[DeviceA-Reth2] member interface gigabitethernet 2/0/2 priority 50
[DeviceA-Reth2] quit
```

3. Create track entries to monitor the link state of the interfaces.

```
[DeviceA] track 1 interface gigabitethernet 1/0/1 physical
[DeviceA-track-1] quit
[DeviceA] track 2 interface gigabitethernet 1/0/2 physical
[DeviceA-track-2] quit
[DeviceA] track 3 interface gigabitethernet 2/0/1 physical
[DeviceA-track-3] quit
[DeviceA] track 4 interface gigabitethernet 2/0/2 physical
[DeviceA-track-4] quit
```

4. Configure a redundancy group:

# Create redundancy group **aaa** and create node 1 for the redundancy group.

```
[DeviceA] redundancy group aaa
[DeviceA-redundancy-group-aaa] node 1
```

# Bind node 1 to Device A.

```
[DeviceA-redundancy-group-aaa-node1] bind slot 1
```

# Set the priority of node 1 to 100.

```
[DeviceA-redundancy-group-aaa-node1] priority 100
```

# Associate track entries 1 and 2 with node 1. Exclude GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 from the shutdown action by the Reth module.

```
[DeviceA-redundancy-group-aaa-node1] track 1 interface gigabitethernet 1/0/1
[DeviceA-redundancy-group-aaa-node1] track 2 interface gigabitethernet 1/0/2
[DeviceA-redundancy-group-aaa-node1] quit
```

# Create node 2 for redundancy group **aaa**.

```
[DeviceA-redundancy-group-aaa] node 2
```

# Bind node 2 to Device B.

```
[DeviceA-redundancy-group-aaa-node2] bind slot 2
```

# Set the priority of node 2 to 50.

```
[DeviceA-redundancy-group-aaa-node2] priority 50
```

# Associate track entries 3 and 4 with node 2. Exclude GigabitEthernet 2/0/1 and GigabitEthernet 2/0/2 from the shutdown action by the Reth module.

```
[DeviceA-redundancy-group-aaa-node2] track 3 interface gigabitethernet 2/0/1
[DeviceA-redundancy-group-aaa-node2] track 4 interface gigabitethernet 2/0/2
[DeviceA-redundancy-group-aaa-node2] quit
```

# Assign Reth 1 and Reth 2 to redundancy group **aaa**.

```
[DeviceA-redundancy-group-aaa] member interface reth 1
[DeviceA-redundancy-group-aaa] member interface reth 2
[DeviceA-redundancy-group-aaa] quit
```

5. Configure settings for routing.

   This example configures static routes, and the next hops in the routes are 1.1.1.1 and 2.2.2.1.

   ```
   [DeviceA] ip route-static 0.0.0.0 0 1.1.1.1
   [DeviceA] ip route-static 3.3.3.0 24 2.2.2.1
   ```

6. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface reth 1
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface reth 2
   [DeviceA-security-zone-Trust] quit
   ```

7. Configure a security policy. Configure a rule named **trust-untrust** to permit the packets between the LAN and the external network.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule 1 name trust-untrust
   [DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
   [DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
   [DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 3.3.3.0 24
   [DeviceA-security-policy-ip-1-trust-untrust] action pass
   [DeviceA-security-policy-ip-1-trust-untrust] quit
   [DeviceA-security-policy-ip] quit
   ```

Verifying the configuration

# Verify that node 1 is the primary node in redundancy group **aaa**.

```
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID   Slot     Priority  Status       Track weight
  Node1     Slot1    100       Primary      255
  Node2     Slot2    50        Secondary    255

Preempt delay time remained   : 0    min
Preempt delay timer setting   : 1    min
Remaining hold-down time      : 0    sec
Hold-down timer setting       : 300  sec
Manual switchover request     : No

Node 1:
  Track info:
    Track     Status        Reduced weight       Interface
    1         Positive      255                  GE1/0/1
    2         Positive      255                  GE1/0/2
Node 2:
  Track info:
    Track     Status        Reduced weight      Interface
    3         Positive      255                 GE2/0/1
    4         Positive      255                 GE2/0/2
```

# Verify that GigabitEthernet 1/0/1 of Reth 1 and GigabitEthernet 1/0/2 of Reth 2 are active.

```
[DeviceA] display reth interface reth 1
Reth1 :
  Redundancy group  : aaa
  Member                Physical status         Forwarding status    Presence status
  GE1/0/1               UP                      Active               Normal
  GE2/0/1               UP                      Inactive             Normal
[DeviceA] display reth interface reth 2
Reth2 :
  Redundancy group  : aaa
  Member                Physical status         Forwarding status    Presence status
  GE1/0/2               UP                      Active               Normal
  GE2/0/2               UP                      Inactive             Normal
```

# Shut down GigabitEthernet 1/0/2.

```
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] shutdown
```

# Verify that node 2 has become the primary node in redundancy group **aaa**.

```
[DeviceA-GigabitEthernet1/0/2] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID   Slot     Priority  Status       Track weight
  Node1     Slot1    100       Secondary    -255
  Node2     Slot2    50        Primary      255

Preempt delay time remained   : 0    min
Preempt delay timer setting   : 1    min
Remaining hold-down time      : 0    sec
Hold-down timer setting       : 300  sec
Manual switchover request     : No

Node 1:
  Track info:
    Track     Status               Reduced weight     Interface
    1         Negative             255                GE1/0/1
    2         Negative(Faulty)     255                GE1/0/2(Fault)
Node 2:
  Track info:
    Track     Status       Reduced weight     Interface
    3         Positive     255                GE2/0/1
    4         Positive     255                GE2/0/2
```

# Verify that GigabitEthernet 1/0/1 has been shut down by the Reth module, and GigabitEthernet 2/0/1 and GigabitEthernet 2/0/2 are active.

```
[DeviceA-GigabitEthernet1/0/2] display reth interface reth 1
Reth1 :
Redundancy group  : aaa
  Member                 Physical status         Forwarding status    Presence status
  GE1/0/1                DOWN(redundancy down)   Inactive             Normal
  GE2/0/1                UP                      Active               Normal
[DeviceA-GigabitEthernet1/0/2] display reth interface reth 2
```

```
Reth2 :
Redundancy group  : aaa
  Member                 Physical status         Forwarding status    Presence status
  GE1/0/2                DOWN                    Inactive             Normal
  GE2/0/2                UP                      Active               Normal
```

NSFOCUS Firewall Series

NF Security Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for security features, including:
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

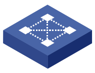
GUI conventions

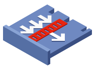
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
(!) **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring security zones

About security zones

You can configure security zones to implement security zone-based security management.

Basic concepts

The security zone feature includes the following basic concepts:

- **Security zone**—A security zone is a collection of interfaces that have the same security requirements.
- **System-defined security zones**—The device provides the following system-defined security zones: **Local**, **Trust**, **DMZ**, **Management**, and **Untrust**. The system creates these security zones automatically when one of following events occurs:
  - The first command for creating a security zone is executed.
  - The first command related to creating an interzone policy is executed.

  System-defined security zones cannot be deleted.
- **DMZ**—A demilitarized zone is a network that is separate from the internal network and the external network both logically and physically. Typically, a DMZ contains devices for the public to access, such as the Web servers and FTP servers.

Security zone-based packet processing rules

The following table describes how the device handles packets when security zone-based security management is configured:

Packets	**Action**
Packets between an interface that is in a security zone and an interface that is not in any security zone	Discard.
Packets between two interfaces that are in the same security zone	Discard by default.
Packets between two interfaces that belong to different security zones	Forward or discard, depending on the matching interzone policy. If no policy is applied or the policy does not exist or does not take effect, the packets are discarded. For more information, see "Creating a zone pair."
Packets between two interfaces that are not in any security zone	Discard.
Packets originated from or destined for the device itself	Forward or discard, depending on the matching interzone policy. By default, these packets are discarded.

Application scenarios

As a best practice, use security zone-based security management if the firewall is connected to multiple network segments or the network topology might change.

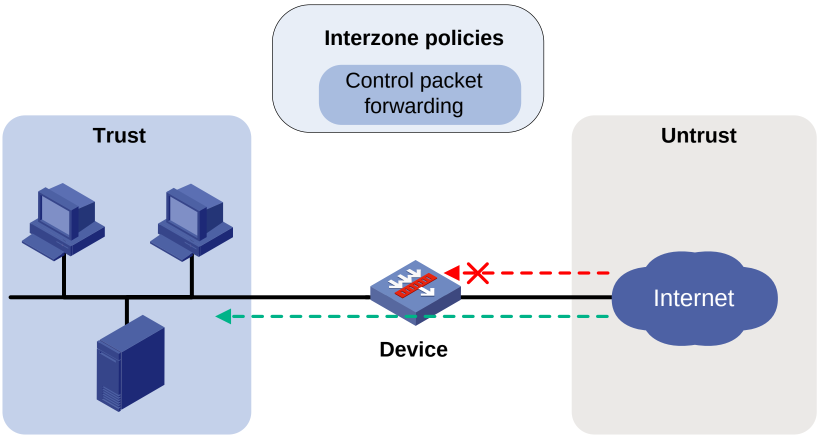
The traditional security management technology is based on interfaces. To filter packets, you must apply interzone policies on the inbound and outbound interfaces of a firewall. When the firewall is connected to multiple network segments, deploying interzone policies is time consuming and complicated. If the network topology changes, you might have to reconfigure interzone policies.

About interzone policies

Interzone policies identify data flows and control packet forwarding based on security zones, as shown in Figure 1. You can assign interfaces with the same security requirements to the same security zone, and assign interfaces with different security requirements to different security zones. For example, you can assign the interface connected to the internal network to security zone **trust**, and assign the interface connected to the Internet to security zone **Untrust**. Then, you can deploy interzone policies between the security zones to control packet forwarding.

If the network topology changes, you only need to change interface assignments. You do not need to modify the interzone policies.

Figure 1 Security zones and interzone policies



Interzone policy types

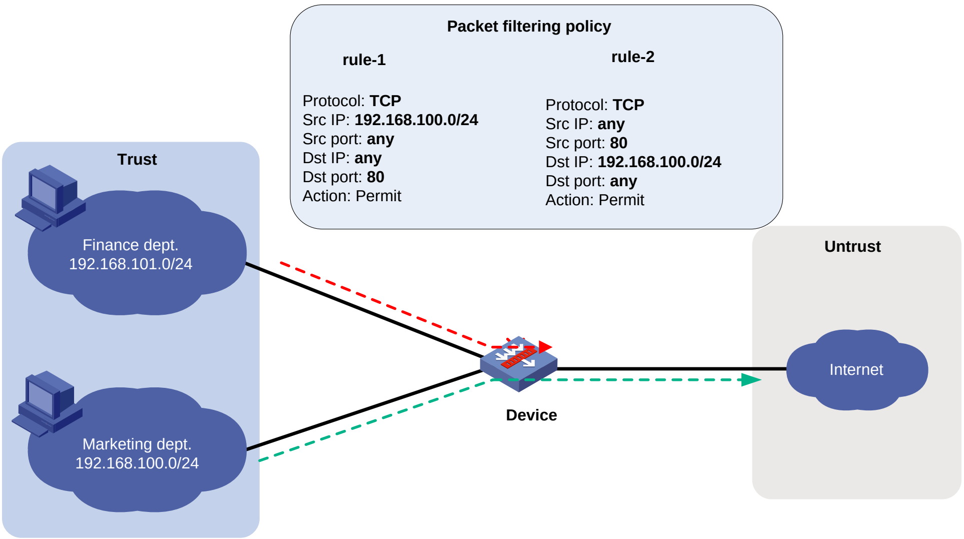
Interzone policies fall in to the following types: packet filtering, ASPF, and security policy. Packet filtering and ASPF can be applied on zone pairs. Security policies are based on security zones and are configured globally.

Packet filtering

Packet filtering controls packet forwarding between security zones by using packet quintuple information, including the source IP address, source port number, destination IP address, destination port number, and protocol number. For more information about packet filtering, see ACL configuration in *ACL and QoS Configuration Guide*.

As shown in Figure 2, to allow only employees in the Marketing department to access the Internet, configure **rule-1** and **rule-2**. The default rule denies access requests from the Finance department. Then, apply packet filtering policies to both directions of the **Trust**-**Untrust** zone pair.

Figure 2 Packet filtering

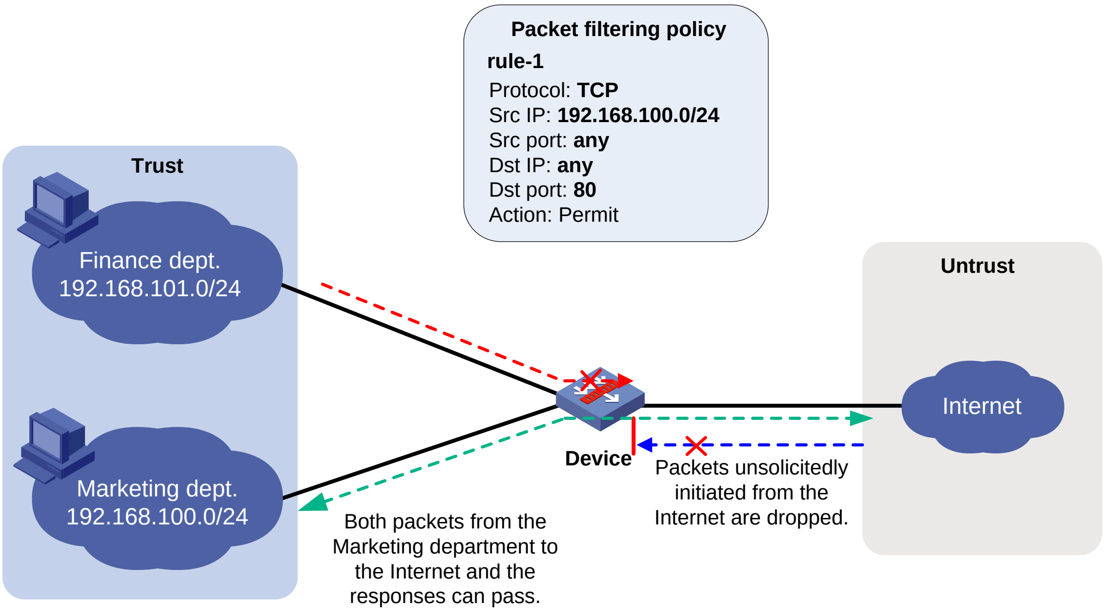


ASPF

Advanced Stateful Packet Filter (ASPF) records information about packets permitted by packet filtering policies to forward responses to the packets between security zones. For more information about ASPF, see "Configuring ASPF."

As shown in Figure 3, to allow only employees in the Marketing department to access the Internet, you can configure **rule-1** and ASPF. The default rule denies access requests from the Finance department but also denies responses to packets from the Marketing department. ASPF enables the device to forward responses to packets from the Marketing department to the Marketing department.

Figure 3 Packet filtering and ASPF

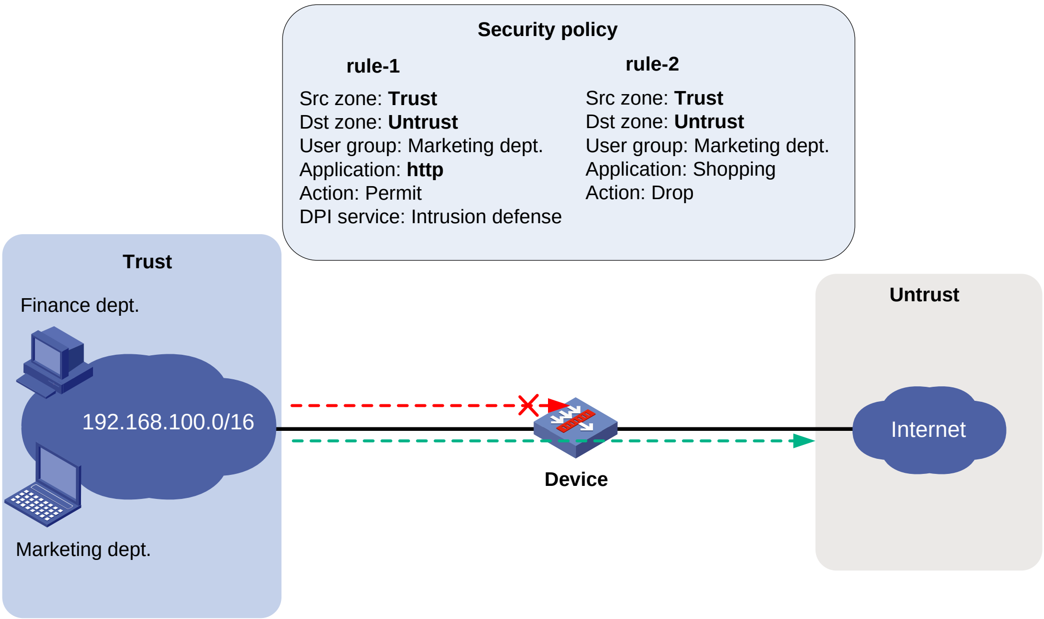


Security policy

Security policies are configured globally and take effect globally. They can be used not only to replace packet filtering, but also to control packet forwarding based on users and applications. Security policies can also use DPI application profiles to perform DPI for matching packets. For more information about security policies, see "Configuring security policies."

As shown in Figure 4, to allow only employees in the Marketing department to access the Internet and forbid access to shopping websites, configure a security policy. The security policy can also implement DPI on web pages. The default rule denies access requests from the Finance department.

Figure 4 Security policy application



Advantages of security policies

As described in "Interzone policy types," security policies have the following advantages over the other types of interzone policies:

- **More flexible and visible management**—Security policies can identify packets based on not only quintuple information but also users.
- **Precise and granular management**—Security policies can identify packets based on protocols (for example, HTTP) and applications (for example, webpage-based games, videos, and shopping).
- **DPI**—By partnering with DPI, it can also provide security protection services such as antivirus and intrusion protection.

Restrictions and guidelines: Security zone configuration

Security policies take precedence over packet filtering policies. Packets matching security policies are processed based on the security policies.

Security zone configuration tasks at a glance

To configure security zones, perform the following tasks:

- Creating a security zone
- Adding members to a security zone

- (Optional.) Specifying a permitted protocol
- Creating a zone pair
- (Optional.) Specifying the default action for packets between interfaces in the same security zone
- (Optional.) Enabling filtering based on virtual service IP address

Creating a security zone

1. Enter system view.
   system-view
2. Create a security zone and enter security zone view.
   security-zone name *zone-name*

   By default, the device has the following security zones: Local, Trust, DMZ, Management, and Untrust.

Adding members to a security zone

About this task

A security zone can include member types listed in Table 1.

Table 1 Security zone members and objects that the members identify

Security zone member	Objects that each member identifies
Layer 3 interface: • Layer 3 Ethernet interface • Layer 3 logical interface, such as a Layer 3 subinterface	All packets received or sent on the interface
Layer 2 interface-VLAN combination	All packets received or sent on the interface that carry the specified VLAN tag
VLAN	All packets that carry the specified VLAN tag. This type of security zones applies only to bridge forwarding. For more information about bridge forwarding, see bridge forwarding configuration in *Layer 2—LAN Switching Configuration Guide.*
IPv4 subnet	All packets sourced from or destined for the IPv4 subnet.
IPv6 subnet	All packets sourced from or destined for the IPv6 subnet.

If a security zone has multiple types of members, a packet is matched in the following order: subnet, interface, and VLAN. The match operation stops when the first matching member is found.

Hardware and feature compatibility

On the default context, only security zone Management has interfaces by default.

The following compatibility matrix shows the default interfaces for security zone Management:

Models	Default interfaces
NFNX5-HD6480, NFNX3-HDB1180, NFNX3-HDB1480	M-GigabitEthernet 1/0/0 M-GigabitEthernet 1/0/1

Models	Default interfaces
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080	M-GigabitEthernet 1/0/0
NFNX3-HDB680, NFNX3-HDB1080	GigabitEthernet 1/0/0 GigabitEthernet 1/0/2

Procedure

1. Enter system view.

   `system-view`

2. Enter security zone view.

   `security-zone name` *`zone-name`*

3. Add members to the security zone.

   Choose one option as needed:

   - Add a Layer 3 Ethernet interface.

     `import interface` *`layer3-interface-type layer3-interface-number`*

     By default, a security zone does not have Layer 3 Ethernet interface members.

     You can perform this step multiple times to add multiple Layer 3 Ethernet interface members.

   - Add Layer 2 interface-VLAN combinations.

     `import interface` *`layer2-interface-type layer2-interface-number`* **`vlan`** *`vlan-list`*

     By default, a security zone does not have Layer 2 interface-VLAN combination members.

     You can perform this step multiple times to add multiple Layer 2 interface-VLAN combination members.

   - Add VLANs.

     `import vlan` *`vlan-list`*

     By default, a security zone does not have VLAN members.

     You can perform this step multiple times to add multiple VLAN members.

   - Add an IPv4 subnet.

     `import ip` *`ip-address`* { *`mask-length`* | *`mask`* } [**`vpn-instance`** *`vpn-instance-name`*]

     By default, a security zone does not have IPv4 subnet members.

     You can perform this step multiple times to add multiple IPv4 subnet members.

   - Add an IPv6 subnet.

     `import ipv6` *`ipv6-address prefix-length`* [**`vpn-instance`** *`vpn-instance-name`*]

     By default, a security zone does not have IPv6 subnet members.

     You can perform this step multiple times to add multiple IPv6 subnet members.

Specifying a permitted protocol

About this task

By default, the device permits packets only between security zones **Local** and **Management**. It denies packets between security zone **Local** and other security zones. To permit packets of the specified protocols between security zone **Local** and other security zones, you can specify permitted protocols on interfaces or configure interzone policies.

After you specify a permitted protocol on an interface, the device will permit packets of the specified protocol from the device that is connected to the interface. The packets will not be limited based on security policies or traffic policies.

You can configure the **`manage`** command multiple times to specify multiple permitted protocols.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specifying a permitted protocol.

   `manage` `{ {` **`http`** `|` **`https`** `|` **`ping`** `|` **`ssh`** `|` **`telnet`** `} {` **`inbound`** `|` **`outbound`** `} | {` **`netconf-http`** `|` **`netconf-https`** `|` **`netconf-ssh`** `|` **`snmp`** `}` **`inbound`** `}`

   By default, no permitted protocols are specified. The device permits packets only from other devices that are connected through interfaces in security zone **Management**.

Creating a zone pair

About this task

A zone pair has a source security zone and a destination security zone. The device examines received first data packets and uses zone pairs to identify data flows. Then, the device uses interzone policies applied to the matching zone pairs to process the data flows.

You can use the **`zone-pair security source any destination any`** command to define the any-to-any zone pair. This zone pair matches all packets from one security zone to another security zone. A zone pair between specific security zones has a higher priority than the any-to-any zone pair.

For packets between the **Management** and **Local** security zones, the device uses only interzone policies applied to the zone pairs of the two security zones. By default, the device forwards packets between the **Management** and **Local** zones.

Procedure

1. Enter system view.

   `system-view`

2. Create a zone pair and enter zone pair view.

   `zone-pair security source` `{` *`source-zone-name`* `|` **`any`** `}` **`destination`** `{` *`destination-zone-name`* `|` **`any`** `}`

Specifying the default action for packets between interfaces in the same security zone

About this task

The system uses the default action for packets that are exchanged between interfaces in the same security zone in the following situations:

- A zone pair from the security zone to the security zone itself is not configured.
- A zone pair from the security zone to the security zone itself is configured, but no interzone policy is applied to the zone pair.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify the default action for packets exchanged between interfaces in the same security zone.
   - Set the default action to **permit**.

     ```
     security-zone intra-zone default permit
     ```

   - Set the default action to **deny**.

     ```
     undo security-zone intra-zone default permit
     ```

   By default, the default action is **deny**.

Enabling filtering based on virtual service IP address

About this task

This feature enables the device to filter packets from external networks to internal servers by virtual service IP address in scenarios where server load balancing is deployed. By default, this feature is disabled. Before matching each of the packets against ACLs, the device translates the destination IP address (the virtual service IP address) to the real server IP address. For more information about packet filtering, see ACL configuration in *ACL and QoS Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable filtering based on virtual service IP address for zone pairs.

   ```
   zone-pair vsip-filter enable
   ```

   By default, filtering based on virtual service IP address is disabled.

Display and maintenance commands for security zones

Execute **display** commands in any view.

Task	Command
Display security zone information.	**display security-zone** [**name** *zone-name*]
Display zone pair information.	**display zone-pair security**

Security zone configuration examples

Example: Configuring security zones

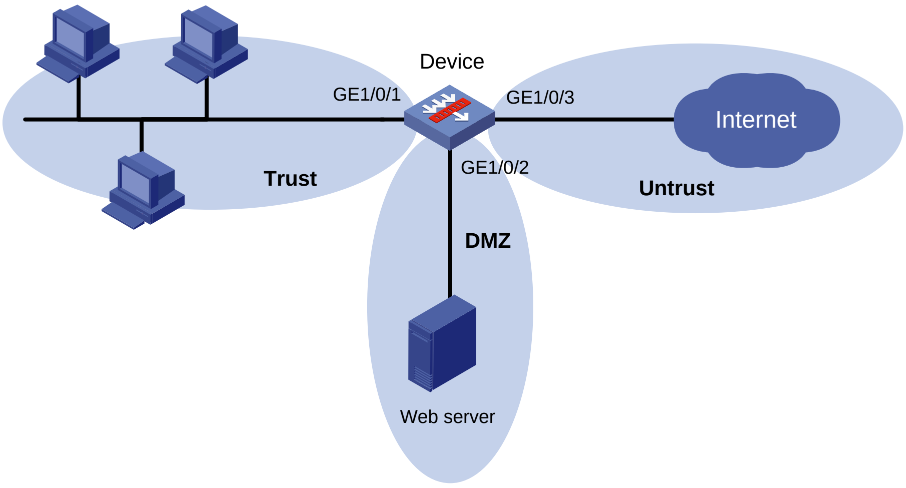
Network configuration

As shown in Figure 5, a security protection device (Device) connects the corporate network to the Internet. The corporate network needs to provide Web services for only internal users.

To ensure corporate network security, configure the device as follows:

- Assign GigabitEthernet 1/0/1, GigabitEthernet 1/0/2, and GigabitEthernet 1/0/3 to security zones **Trust**, **DMZ**, and **Untrust**, respectively.
- Configure zone pairs and apply interzone policies to control access as follows:
  - Allow internal users to access the Web server and the Internet.
  - Forbid external users to access the internal network and the Web server.
  - Forbid the Web server to access the internal network.

Figure 5 Network diagram



Procedure

1. Assign IP address to the interfaces.

   # Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   ```

```
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-security-zone-Untrust] quit
```

3. Configure ACLs.

# Configure ACL 3001 to allow internal hosts to access the Internet.

```
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3500] rule permit ip source 1.1.1.0 0.0.0.255 destination
3.3.3.0 0.0.0.255
[Device-acl-ipv4-adv-3500] quit
```

# Configure ACL 3002 to allow internal hosts to access the Web server.

```
[Device] acl advanced 3002
[Device-acl-ipv4-adv-3002] rule permit ip source 1.1.1.0 0.0.0.255 destination
2.2.2.0 0.0.0.255
[Device-acl-ipv4-adv-3500] quit
```

4. Configure zone pairs.

# Create a zone pair with the source security zone **Trust** and destination security zone **Untrust**. Apply ACL 3001 to the zone pair.

```
[Device] zone-pair security source trust destination untrust
[Device-zone-pair-security-Trust-Untrust] packet-filter 3001
[Device-zone-pair-security-Trust-Untrust] quit
```

# Create a zone pair with the source security zone **Trust** and destination security zone **DMZ**. Apply ACL 3002 to the zone pair.

```
[Device] zone-pair security source trust destination dmz
[Device-zone-pair-security-Trust-DMZ] packet-filter 3002
[Device-zone-pair-security-Trust-DMZ] quit
```

Verifying the configuration

# Verify that internal hosts can access the Internet and the Web server. (Details not shown.)

# Verify that access requests initiated from the Internet and the **DMZ** zone to the internal network are denied. (Details not shown.)

Contents

Configuring security policies

About security policies

A security policy defines a set of rules for forwarding control and Deep Packet Inspection (DPI). It matches packets against the rules and takes the action stated in the rules on the matched packets.

Security policy rules

A security policy contains one or multiple rules. Each security policy rule is a permit or deny, or DPI statement for identifying traffic based on criteria.

Rule numbering

Each rule is uniquely identified by a name and an ID. When you create a rule, the rule name must be manually configured, and the rule ID can be manually configured or automatically assigned by the system.

Rule match criteria

The rule match criteria include the following types: source security zone, destination security zone, source IP address and source MAC address, destination IP address, user and user group, application and application group, VPN instance, and service.

You can specify multiple criteria for each type, except VPN instance. For example, you can configure multiple source security zones for a rule.

Rule and session management

When a security policy is configured, the device generates session entries for permitted packets to record packet information.

You can set session aging times for protocol states, application layer protocols, or rules. The aging time configured for a rule takes precedence over the aging time configured for a protocol state or an application layer protocol. For more information about session management, see "Managing sessions."

Security policy mechanism

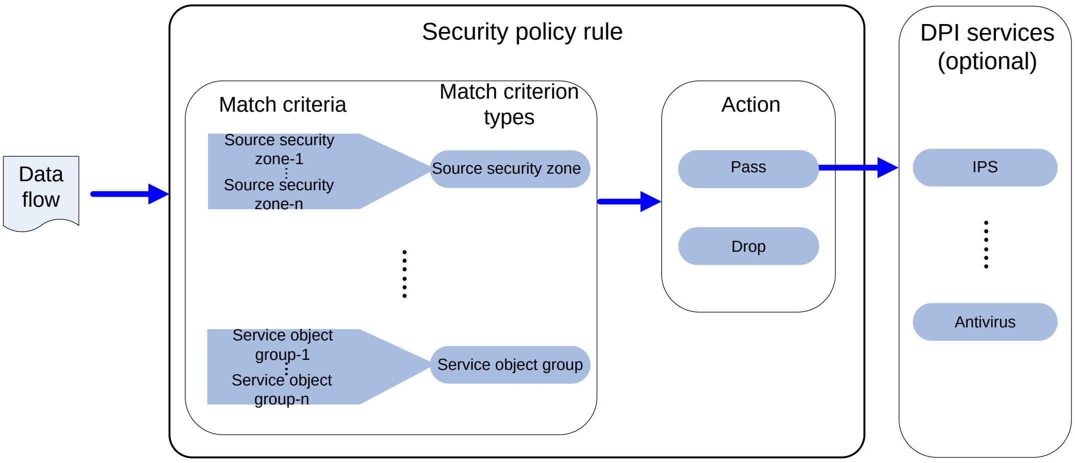
As shown in Figure 1, a security policy operates as follows:

1. After receiving a packet, the device matches the packet against the configured security policy rules.

   A security policy rule includes various match criterion types. A packet is considered matched if it matches all the criterion types in the rule. Each criterion type includes one or more criteria, and a packet matches a criterion type if it matches any criterion of the type. Source MAC address criteria and source IP address criteria belong to the same criterion type.

   - If no match is found, the device discards the packet.
   - If a match is found and the rule action is **drop**, the device discards the packet.
   - If a match is found and the rule action is **pass**, the device goes to the next step.
2. If a DPI application profile is configured for the matched rule, the device uses the specified profile to perform DPI on the packet. If no DPI application profile is specified, the device allows the packet to pass.

Figure 1 Security policy mechanism



Rule matching acceleration

This feature accelerates security policy rule matching to enhance connection establishment and packet forwarding performance, especially for a device using multiple rules to match packets from multiple users.

Security policy rule grouping

Security policy rule grouping allows users to enable, disable, delete, and move security policy rules in batches. A security policy rule in a security policy rule group takes effect only when both the rule and the group are enabled.

Application scenarios

Local device access

In a local access scenario, you must configure security policy rules for the following operations to be performed successfully:

- Local device access through a non-management local port, including ping, telnet, Web access, and FTP operations.
- Protocol packet exchange triggered by any other devices (such as packet exchanges of Dynamic Routing Protocol or VPN tunneling).

For example, as shown in Figure 2, for a PC (10.1.1.10) in the Trust security zone to access the Web interface of the local device (10.1.1.1) through HTTPS, you must configure a security policy rule as described in Table 1.

Figure 2 Local device access

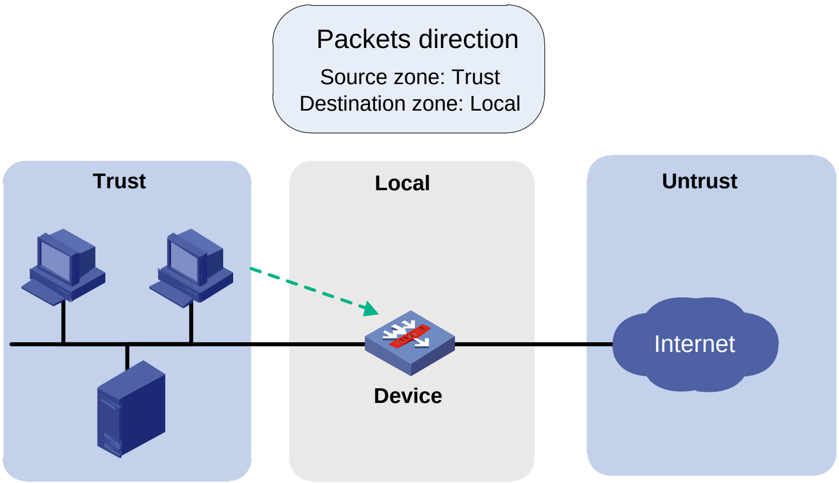


Table 1 Security policy rule configuration

Rule name	Source security zone	Destination security zone	Source IP address	Destination IP address	Service	Action
httpslocalin	Trust	Local	10.1.1.10	10.1.1.1	https	pass

For more information about permitting protocol packets, see "Common local services."

Remote device access

In a remote access scenario, you must configure security policy rules for the following operations to be performed successfully:

- Remote device access through a non-management local port, including ping, telnet, Web access, and FTP operations.
- Protocol packet exchange triggered by the local device (such as packet exchanges of Dynamic Routing Protocol or VPN tunneling).

For example, as shown in Figure 3, for the local device (10.1.1.1) to access the FTP service provided by the FTP server (10.1.1.20) in the Trust security zone, you must configure a security policy rule as described in Table 2.

Figure 3 Remote device access

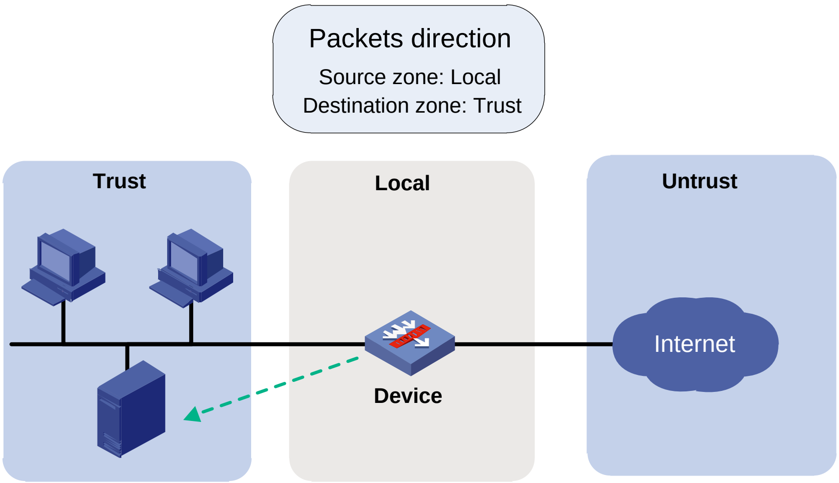


Table 2 Configuration of local device accessing other devices

Rule name	Source security zone	Destination security zone	Source IP address	Destination IP address	Service	Action
ftplocalout	Local	Trust	10.1.1.1	10.1.1.20	ftp	pass

Direct traffic forwarding

For the local device to forward traffic that is either sent from or destined to the local device, you must configure security policy rules to permit traffic from the corresponding security zones.

For example, as shown in Figure 4, for a PC (10.1.1.10) in the Trust zone to visit a website in the Untrust zone through HTTP, you must configure a security policy rule as shown in Table 3.

Figure 4 Direct traffic forwarding

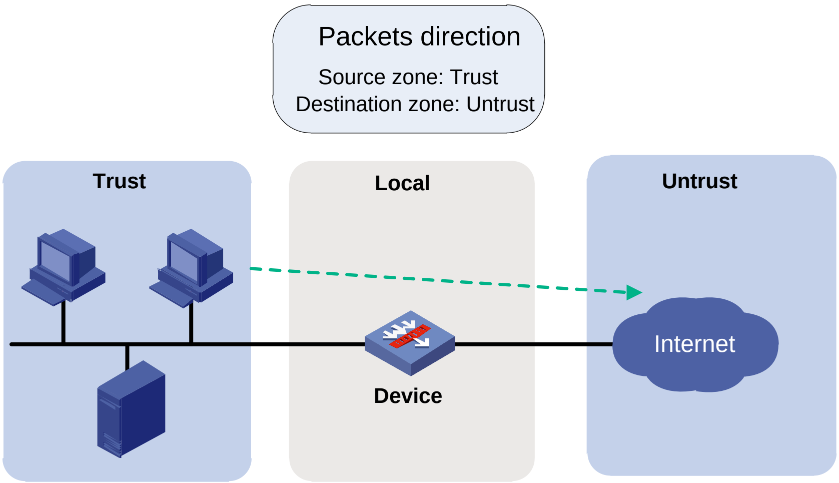


Table 3 Direct traffic forwarding

Rule name	Source security zone	Destination security zone	Source IP address	Service	Action
trust-untrust	Trust	Untrust	10.1.1.10	http	pass

Configuration principles

As a best practice to achieve the optimal performance, follow these principles when you configure security policy rules:

- Configure matching criteria as strict as possible.
- Follow the depth-first order during rule creation to create rules with stricter match criteria first because the system matches packets against rules in the order the rules were created.
- For packets from and to the same security zone, configure two rules to control packet exchanges between interfaces instead of using one rule that allows traffic between any interfaces in the zone to pass.
- Place security policy rules using the Local security zone as a matching criterion in front of all the other rules to ensure that local service packets can be correctly processed.

Common local services

You must configure security policy rules to permit local service traffic from or to specific security zones. Table 4 lists the common local services that require security policy permission. Support for the common local services varies by device.

Table 4 Common local services that require security policy permission

Service	Required security policy configuration
OSPF/IS-IS/RIP/BGP	Configure rules for the service traffic from or to the local zone.
IPsec/SSL/L2TP/MPLS/GRE	Configure rules for the service traffic from or to the local zone.
DNS/DHCP/FTP (client)	Configure rules only for the service traffic from the local zone.
DNS/DHCP/FTP (server)	Configure rules only for the service traffic to the local zone.
SSH/Telnet/Ping/Tracet (locally triggered)	Configure rules only for the service traffic from the local zone.
SSH/Telnet/SNMP/HTTP/HTTPS (local access)	Configure rules only for the service traffic to the local zone.
BFD	Configure rules for the service traffic from or to the local zone.
LB	Configure rules for the service traffic from or to the local zone.

Restrictions and guidelines: Security policy configuration

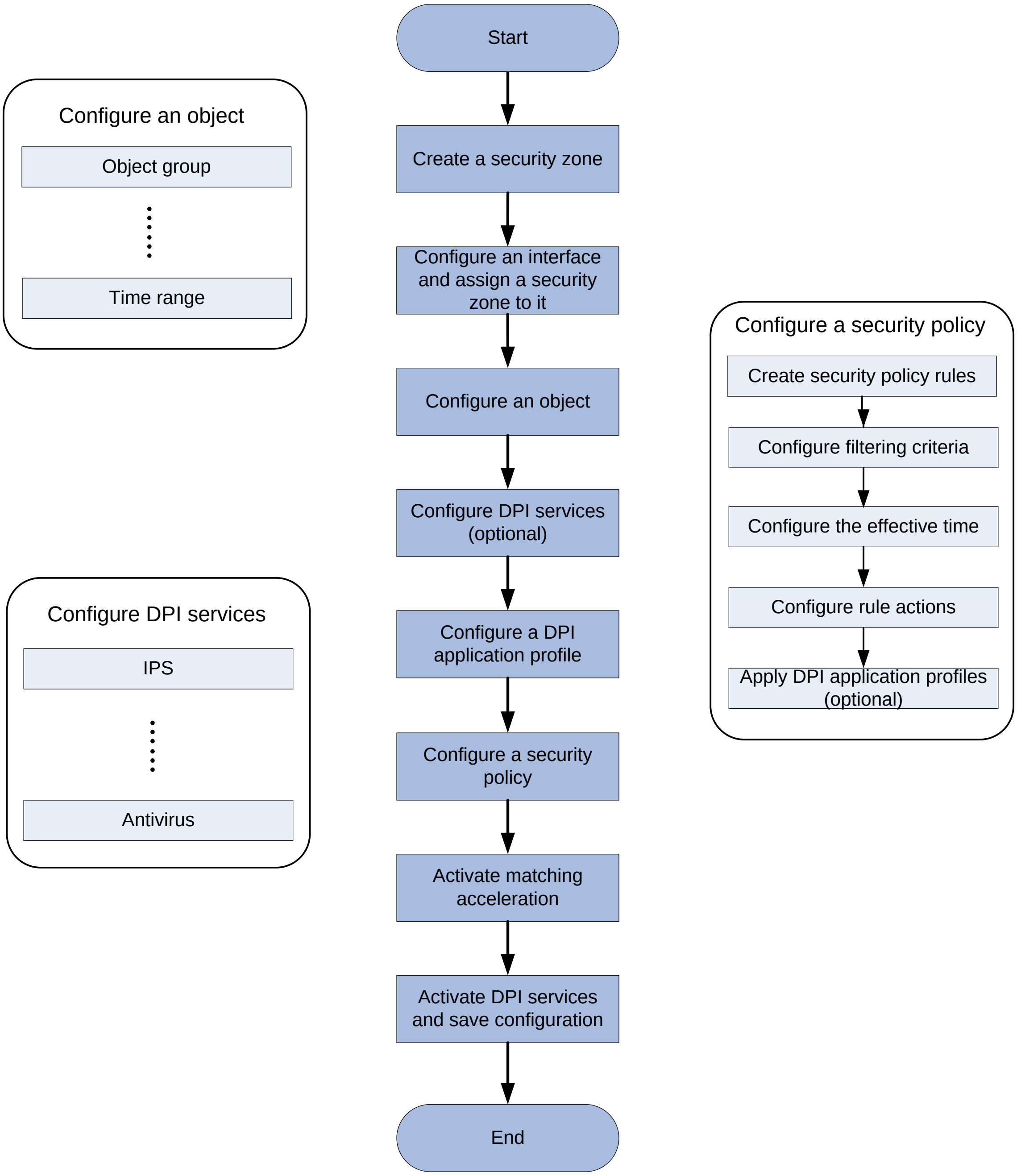
As a best practice, do not configure packet filtering and object policies at the same time. If you do so, some policies might fail to take effect, causing service interruption.

Before configuring security policies, verify if the device is configured with packet filtering. As a best practice, switch packet filtering and object policies to security policies because security policies take precedence over packet filtering. For more information, see "Switching packet filtering to security policy settings."

Configuration procedure diagram

Figure 5 shows how to configure a security policy.

Figure 5 Security policy configuration procedure


Start
Create a security zone
Configure an interface and assign a security zone to it
Configure an object
Configure DPI services (optional)
Configure a DPI application profile
Configure a security policy
Activate matching acceleration
Activate DPI services and save configuration
End
Configure an object
Object group
Time range
Configure DPI services
IPS
Antivirus
Configure a security policy
Create security policy rules
Configure filtering criteria
Configure the effective time
Configure rule actions
Apply DPI application profiles (optional)

Prerequisites for security policies

Before you configure security policies, perform the following tasks:

- Configure a time range. See time range configuration in *ACL and QoS Configuration Guide*.
- Configure IP address object groups and service object groups. See "Configuring object groups."
- Configure applications and application groups. See "Configuring APR."
- Configure user and user groups. See "Configuring user identification."
- Configure security zones. See "Configuring security zones.".

- Configure DPI. See *DPI Configuration Guide*.

Security policy tasks at a glance

To configure object policies, perform the following tasks:

1. (Optional.) Switching packet filtering to security policy settings
2. Enabling the security policy feature
3. Configuring IPv4 security policy rules
   a. Creating a security policy rule
   b. Configuring filtering criteria for a security policy rule
   c. Specifying the action for a security policy rule
   d. (Optional.) Specifying a time range for a security policy rule
   e. (Optional.) Applying a DPI application profile to a security policy rule
   f. (Optional.) Setting the session aging time for a security policy rule
   g. (Optional.) Associating a security policy rule with a track entry
   h. (Optional.) Enabling logging for matched packets
   i. (Optional.) Enabling statistics collection for matched packets
4. Configuring IPv6 security policy rules
   a. Creating a security policy rule
   b. Configuring filtering criteria for a security policy rule
   c. Specifying the action for a security policy rule
   d. (Optional.) Specifying a time range for a security policy rule
   e. (Optional.) Applying a DPI application profile to a security policy rule
   f. (Optional.) Setting the session aging time for a security policy rule
   g. (Optional.) Associating a security policy rule with a track entry
   h. (Optional.) Enabling logging for matched packets
   i. (Optional.) Enabling statistics collection for matched packets
5. (Optional.) Manage security policies
   a. Changing the rule match order
   b. Activating rule matching acceleration
   c. Disabling a security policy rule
   d. Renaming a security policy rule
6. (Optional.) Configuring security policy rule groups
   a. Creating a security policy rule group
   b. Specifying a security policy rule group for a security policy rule
   c. Moving a security policy rule group
   d. Renaming a security policy rule group
7. (Optional.) Setting the time for fast output of security policy settings as logs

Switching packet filtering to security policy settings

When security policies are configured, packet filtering is performed only on packets that do not match any security policy rule. As a best practice to avoid interruption of services permitted by packet filtering, do not configure packet filtering and security policies at the same time.

If packet filtering is configured, switch packet filtering to security policy settings as a best practice before enabling the security policy feature.

Switching packet filtering to security policy settings before security policies are configured

1. Execute the **`security-policy disable`** command to disable the security policy feature.
2. Enter security policy view, configure filtering criteria and actions for security policy rules based on the packet filtering configuration.
3. Execute the **`undo security-policy disable`** command to enable the security policy feature.
4. (Optional.) Delete the packet filtering configuration from the device.

Switching packet filtering to security policy settings after security policies are configured

You can perform this task from the Web interface (recommended) or CLI.

To perform this task from the CLI:

1. Access the CLI of the device. As a best practice, access the device through the Console port or interface of the Management security zone in case the connection is terminated by packet filtering.
2. Enter security policy view, and configure filtering criteria and actions for security policy rules based on the packet filtering configuration.

> △ **CAUTION:**
>
> After using the **`rule`** [*`rule-id`*] **name** *`rule-name`* command to create the first security policy rule, configure the filtering criteria and actions based on packet filtering as soon as possible to shorten service interruption. With no filtering criteria or action specified, the rule matches and drops all packets. You can configure other security policy settings when the service is not busy as a best practice.

3. (Optional.) Delete the packet filtering configuration from the device.

Enabling the security policy feature

Restrictions and guidelines

Security policy settings take effect only when the security policy feature is enabled.

Procedure

1. Enter system view.

   `system-view`

2. Enable the security policy feature.

   undo security-policy disable

   By default, the security policy feature is enabled.

△ **CAUTION:**

The **security-policy disable** command disables the security policy feature and might cause traffic interruption.

Configuring IPv4 security policy rules

Creating a security policy rule

About this task

By default, no rules exist in a security policy, and the device allows only packets exchanged between the Management security zone and the Local security zone to pass. For the device to process packets correctly, configure policy rules for each security policy.

If a configured feature, such as dynamic routing, requires exchanges with the device, configure security policy rules for the Local security zone to communicate with the specific zones.

Procedure

1. Enter system view.

   system-view

2. Enter IPv4 security policy view.

   security-policy ip

△ **CAUTION:**

- The **undo security-policy** { **ip** | **ipv6** } command directly deletes all security policy configurations and might cause network interruptions.

3. (Optional.) Configure a description for the policy.

   description *text*

   By default, a security policy does not have a description.

4. Create a security policy rule.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

5. (Optional.) Configure a description for the rule.

   description *text*

   By default, a security policy rule does not have a description.

Configuring filtering criteria for a security policy rule

Restrictions and guidelines

A rule matches all packets if no criteria are specified for the rule. If no action is set for the rule, the device drops the matched packets by default.

If a specified object group has no objects, the rule cannot match any packets.

Packets exchanged between the Management and Local security zones are allowed to pass by default and can only match local-to-management or management-to-local security policy rules. To discard packets between the Management and Local security zones, configure local-to-management and management-to-local rules and specify the rule actions as **drop**.

Security policy rules specified with an IP address object group that uses a user or user group cannot match packets. To filter packets by user or user group, configure security policy rules specified with user or user group criteria.

Procedure

1. Enter system view.

   system-view

2. Enter IPv4 security policy view.

   security-policy ip

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Configure source filtering criteria:
   - Specify a source security zone as a filtering criterion.

     source-zone *source-zone-name*

     By default, no source security zone is specified as a filtering criterion.
   - Specify a source IPv4 address object group as a filtering criterion.

     source-ip *object-group-name*

     By default, no source IPv4 address object group is specified as a filtering criterion.
   - Specify a source IPv4 host address as a filtering criterion.

     source-ip-host *ip-address*

     By default, no source IPv4 host address is specified as a filtering criterion.
   - Specify a source IPv4 subnet as a filtering criterion.

     source-ip-subnet *ip-address* { *mask-length* | *mask* }

     By default, no source IPv4 subnet is specified as a filtering criterion.
   - Specify a source IPv4 address range as a filtering criterion.

     source-ip-range *ip-address1 ip-address2*

     By default, no source IPv4 address range is specified as a filtering criterion.
   - Specify a source MAC address object group as a filtering criterion.

     source-mac *object-group-name*

     By default, no source MAC address object group is specified as a filtering criterion.

5. Configure destination filtering criteria:
   - Specify a destination security zone as a filtering criterion.

     destination-zone *destination-zone-name*

     By default, no destination security zone is specified as a filtering criterion.
   - Specify a destination IPv4 address object group as a filtering criterion.

     destination-ip *object-group-name*

     By default, no destination IPv4 address object group is specified as a filtering criterion.
   - Specify a destination IPv4 host address as a filtering criterion.

     destination-ip-host *ip-address*

     By default, no destination IPv4 host address is specified as a filtering criterion.
   - Specify a destination IPv4 subnet as a filtering criterion.

     destination-ip-subnet *ip-address* { *mask-length* | *mask* }

     By default, no destination IPv4 subnet is specified as a filtering criterion.

- Specify a destination IPv4 address range as a filtering criterion.

  destination-ip-range *ip-address1 ip-address2*

  By default, no destination IPv4 address range is specified as a filtering criterion.

6. Specify a service object group as a filtering criterion.

   service { *object-group-name* | **any** }

   By default, no service object group is specified as a filtering criterion.

7. Specify a service port as a filtering criterion.

   service-port *protocol* [{ **destination** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } | **source** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } } * | *icmp-type icmp-code* | *icmpv6-type icmpv6-code*]

   By default, no service port is specified as a filtering criterion.

8. Configure application filtering criteria:
   - Specify an application as a filtering criterion.

     application *application-name*

     By default, no application is specified as a filtering criterion.

     For the application filtering criteria to be identified, you must permit the dependent applications to pass through.
   - Specify an application group as a filtering criterion.

     app-group *app-group-name*

     By default, no application group is specified as a filtering criterion.

9. Configure user filtering criterion:
   - Specify a user as a filtering criterion.

     user *username* [**domain** *domain-name*]

     By default, no user is specified as a filtering criterion.
   - Specify a user group as a filtering criterion.

     user-group *user-group-name* [**domain** *domain-name*]

     By default, no user group is specified as a filtering criterion.

10. Configure the rule to take effect on received packets of the specified VPN instance.

    vrf *vrf-name*

    By default, a security policy rule takes effect on received packets of the public network.

Specifying the action for a security policy rule

1. Enter system view.

   system-view

2. Enter IPv4 security policy view.

   security-policy ip

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Specify the action for the security policy rule.

   action { **drop** | **pass** }

   By default, the action for a security policy rule is **drop**.

Specifying a time range for a security policy rule

1. Enter system view.

   `system-view`

2. Enter IPv4 security policy view.

   `security-policy ip`

3. Enter security policy rule view.

   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }

4. Specify a time range during which the security policy rule is in effect.

   `time-range` *`time-range-name`*

   By default, a security policy rule is in effect at any time.

Applying a DPI application profile to a security policy rule

About this task

This feature enables the device to perform DPI on packets matching the specified rule. For more information about DPI, see *DPI Configuration Guide*.

Restrictions and guidelines

This feature takes effect only when the rule action is **pass**.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv4 security policy view.

   `security-policy ip`

3. Enter security policy rule view.

   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }

4. Specify the rule action as **pass**.

   `action` **pass**

   By default, the action for a security policy rule is **drop**.

5. Apply a DPI application profile to the rule.

   `profile` *`app-profile-name`*

   By default, no DPI application profile is applied to a rule.

Setting the session aging time for a security policy rule

About this task

Perform this task to specify the aging time for stable sessions and persistent sessions. The configuration takes effect only on sessions established afterwards.

The configured aging time for persistent sessions is effective only on TCP sessions in ESTABLISHED state.

The priorities of the session aging times configured by using the **`session persistent aging-time`**, **`session aging-time`**, and **`session persistent acl`** commands are in descending order.

Procedure

1. Enter system view.

   system-view

2. Enter IPv4 security policy view.

   security-policy ip

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Set the session aging time.

   session aging-time *time-value*

   By default, the session aging time is not configured.

5. Set the aging time for persistent sessions.

   session persistent aging-time *time-value*

   By default, the aging time for persistent sessions is not configured.

△ **CAUTION:**

Setting too long an aging time might cause persistent sessions to increase rapidly and therefore cause the CPU usage to be high.

Associating a security policy rule with a track entry

About this task

Perform this task to enable the collaboration between the track module and a security policy rule. The collaboration operates as follows:

- If a rule is associated with the Negative state of a track entry, the device:
  - Sets the rule state to Active if the track entry is in Negative state.
  - Sets the rule state to Inactive if the track entry is in Positive state.
- If a rule is associated with the Positive state of a track entry, the device:
  - Sets the rule state to Active if the track entry is in Positive state.
  - Sets the rule state to Inactive if the track entry is in Negative state.

For more information about track entries, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter IPv4 security policy view.

   security-policy ip

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Associate the rule with a track entry.

   track { **negative** | **positive** } *track-entry-number*

   By default, no track entry is associated with a rule.

Enabling logging for matched packets

About this task

This feature enables the device to log matching packets and send the log to the information center for processing or fast output the log. The log destinations and output rules are determined by the information center or fast log output settings. For more information about the information center and fast log output, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv4 security policy view.

   `security-policy ip`

3. Enter security policy rule view.

   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }

4. Enable logging for matched packets.

   `logging enable`

   By default, logging for matched packets is disabled.

Enabling statistics collection for matched packets

About this task

Perform this task to enable the device to collect statistics about matched packets. The collected statistics can be viewed by executing the **`display security-policy statistics`** command.

If an enabling period is specified, the system disables the statistics collection feature and removes the configuration at period expiration. If no enabling period is specified, you must execute the **undo** command to disable the statistics collection feature.

Restrictions and guidelines

When inter-VLAN bridge forwarding is configured, this feature collects statistics only about packets discarded by security policies and DPI. Statistics about permitted packets are not collected. For more information about inter-VLAN bridge forwarding, see Layer 2 forwarding in *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv4 security policy view.

   `security-policy ip`

3. Enter security policy rule view.

   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }

4. Enable statistics collection for matched packets.

   `counting enable` [**`period`** *`value`*]

   By default, the device does not collect statistics about matched packets.

Configuring IPv6 security policy rules

Creating a security policy rule

About this task

By default, no rules exist in a security policy, and the device allows only packets exchanged between the Management security zone and the Local security zone to pass. For the device to process packets correctly, configure policy rules for each security policy.

If a configured feature, such as dynamic routing, tunneling, and VPN, requires exchanges with the device, configure security policy rules for the Local security zone to communicate with the specific zones.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   `security-policy ipv6`

> **CAUTION:**
>
> - The **`undo security-policy`** { **`ip`** | **`ipv6`** } command directly deletes all security policy configurations and might cause network interruptions.

3. (Optional.) Configure a description for the policy.

   `description` *`text`*

   By default, a security policy does not have a description.

4. Create a security policy rule.

   `rule` { *`rule-id`* | [*`rule-id`*] **`name`** *`rule-name`* }

5. (Optional.) Configure a description for the rule.

   `description` *`text`*

   By default, a security policy rule does not have a description.

Configuring filtering criteria for a security policy rule

Restrictions and guidelines

A rule matches all packets if no criteria are specified for the rule. If no action is set for the rule, the device drops the matched packets by default.

If a specified object group has no objects, the rule cannot match any packets.

Packets exchanged between the Management and Local security zones are allowed to pass by default and can only match local-to-management or management-to-local security policy rules. To discard packets between the Management and Local security zones, configure local-to-management and management-to-local rules and specify the rule actions as **drop**.

Security policy rules specified with an IP address object group that uses a user or user group cannot match packets. To filter packets by user or user group, configure security policy rules specified with user or user group criteria.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   security-policy ipv6

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Configure source filtering criteria:
   - Specify a source security zone as a filtering criterion.

     source-zone *source-zone-name*

     By default, no source security zone is specified as a filtering criterion.
   - Specify a source IPv6 address object group as a filtering criterion.

     source-ip *object-group-name*

     By default, no source IPv6 address object group is specified as a filtering criterion.
   - Specify a source IPv6 host address as a filtering criterion.

     source-ip-host *ipv6-address*

     By default, no source IPv6 host address is specified as a filtering criterion.
   - Specify a source IPv6 subnet as a filtering criterion.

     source-ip-subnet *ipv6-address prefix-length*

     By default, no source IPv6 subnet is specified as a filtering criterion.
   - Specify a source IPv6 address range as a filtering criterion.

     source-ip-range *ipv6-address1 ipv6-address2*

     By default, no source IPv6 address range is specified as a filtering criterion.

5. Configure destination filtering criteria:
   - Specify a destination security zone as a filtering criterion.

     destination-zone *destination-zone-name*

     By default, no destination security zone is specified as a filtering criterion.
   - Specify a destination IPv6 address object group as a filtering criterion.

     destination-ip *object-group-name*

     By default, no destination IPv6 address object group is specified as a filtering criterion.
   - Specify a destination IPv6 host address as a filtering criterion.

     destination-ip-host *ipv6-address*

     By default, no destination IPv6 host address is specified as a filtering criterion.
   - Specify a destination IPv6 subnet as a filtering criterion.

     destination-ip-subnet *ipv6-address prefix-length*

     By default, no destination IPv6 subnet is specified as a filtering criterion.
   - Specify a destination IPv6 address range as a filtering criterion.

     destination-ip-range *ipv6-address1 ipv6-address2*

     By default, no destination IPv6 address range is specified as a filtering criterion.

6. Specify a service object group as a filtering criterion.

   service { *object-group-name* | **any** }

   By default, no service object group is specified as a filtering criterion.

7. Specify a service port as a filtering criterion.

   service-port *protocol* [{ **destination** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } | **source** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } } * | *icmpv6-type icmpv6-code*]

   By default, no service port is specified as a filtering criterion.

8. Configure application filtering criteria:
   - Specify an application as a filtering criterion.

     application *application-name*

     By default, no application is specified as a filtering criterion.

     For the application filtering criteria to be identified, you must permit the dependent applications to pass through.
   - Specify an application group as a filtering criterion.

     app-group *app-group-name*

     By default, no application group is specified as a filtering criterion.
9. Configure user filtering criterion:
   - Specify a user as a filtering criterion.

     user *username* [**domain** *domain-name*]

     By default, no user is specified as a filtering criterion.
   - Specify a user group as a filtering criterion.

     user-group *user-group-name* [**domain** *domain-name*]

     By default, no user group is specified as a filtering criterion.
10. Configure the rule to take effect on received packets of the specified VPN instance.

    vrf *vrf-name*

    By default, a security policy rule takes effect on received packets of the public network.

Specifying the action for a security policy rule

1. Enter system view.

   system-view
2. Enter IPv6 security policy view.

   security-policy ipv6
3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }
4. Specify the action for the security policy rule.

   action { **drop** | **pass** }

   By default, the action for a security policy rule is **drop**.

Specifying a time range for a security policy rule

1. Enter system view.

   system-view
2. Enter IPv6 security policy view.

   security-policy ipv6
3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }
4. Specify a time range during which the security policy rule is in effect.

   time-range *time-range-name*

   By default, a security policy rule is in effect at any time.

Applying a DPI application profile to a security policy rule

About this task

This feature enables the device to perform DPI on packets matching the specified rule. For more information about DPI, see *DPI Configuration Guide*.

Restrictions and guidelines

This feature takes effect only when the rule action is **pass**.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   `security-policy ipv6`

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Specify the rule action as **pass**.

   action pass

   By default, the action for a security policy rule is **drop**.

5. Apply a DPI application profile to the rule.

   profile *app-profile-name*

   By default, no DPI application profile is applied to a rule.

Setting the session aging time for a security policy rule

About this task

Perform this task to specify the aging time for stable sessions and persistent sessions. The configuration takes effect only on sessions established afterwards.

The configured aging time for persistent sessions is effective only on TCP sessions in ESTABLISHED state.

The priorities of the session aging times configured by using the **session persistent aging-time**, **session aging-time**, and **session persistent acl** commands are in descending order.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   `security-policy ipv6`

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Set the session aging time.

   session aging-time *time-value*

   By default, the session aging time is not configured.

5. Set the aging time for persistent sessions.

   session persistent aging-time *time-value*

By default, the aging time for persistent sessions is not configured.

CAUTION:

Setting too long an aging time might cause persistent sessions to increase rapidly and therefore cause the CPU usage to be high.

Associating a security policy rule with a track entry

About this task

Perform this task to enable the collaboration between the track module and a security policy rule. The collaboration operates as follows:

- If a rule is associated with the Negative state of a track entry, the device:
  - Sets the rule state to Active if the track entry is in Negative state.
  - Sets the rule state to Inactive if the track entry is in Positive state.
- If a rule is associated with the Positive state of a track entry, the device:
  - Sets the rule state to Active if the track entry is in Positive state.
  - Sets the rule state to Inactive if the track entry is in Negative state.

For more information about track entries, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   `security-policy ipv6`

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Associate the rule with a track entry.

   track { **negative** | **positive** } *track-entry-number*

   By default, no track entry is associated with a rule.

Enabling logging for matched packets

About this task

This feature enables the device to log matching packets and send the log to the information center for processing or fast output the log. The log destinations and output rules are determined by the information center or fast log output settings. For more information about the information center and fast log output, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv6 security policy view.

   `security-policy ipv6`

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Enable logging for matched packets.

logging enable

By default, logging for matched packets is disabled.

Enabling statistics collection for matched packets

About this task

Perform this task to enable the device to collect statistics about matched packets. The collected statistics can be viewed by executing the **display security-policy statistics** command.

If an enabling period is specified, the system disables the statistics collection feature and removes the configuration at period expiration. If no enabling period is specified, you must execute the **undo** command to disable the statistics collection feature.

Restrictions and guidelines

When inter-VLAN bridge forwarding is configured, this feature collects statistics only about packets discarded by security policies and DPI. Statistics about permitted packets are not collected. For more information about inter-VLAN bridge forwarding, see Layer 2 forwarding in *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter IPv6 security policy view.

   security-policy ipv6

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Enable statistics collection for matched packets.

   counting enable [**period** *value*]

   By default, the device does not collect statistics about matched packets.

Changing the rule match order

About this task

The device matches packets against security policy rules in the order the rules were created. You can change the rule match order by changing the position of a security policy rule in the rule list.

Procedure

1. Enter system view.

   system-view

2. Enter IPv4 or IPv6 security policy view.

   security-policy { **ip** | **ipv6** }

3. Move a security policy rule.

   - Move a security policy rule by rule ID.

     move rule *rule-id1* { { **after** | **before** } *rule-id2* | **bottom** | **down** | **top** | **up** }

   - Move a security policy rule by rule name.

     move rule name *rule-name1* { { **after** | **before** } **name** *rule-name2* | **bottom** | **down** | **top** | **up** }

Activating rule matching acceleration

About this task

Rule matching acceleration does not take effect on newly added, modified, and moved rules unless the feature is activated for the rules. By default, the system automatically activates rule matching acceleration for such rules at specific intervals. The interval is 2 seconds if 100 or fewer rules exist and 20 seconds if over 100 rules exist.

To activate rule matching acceleration immediately after a rule change, you can perform this task.

Restrictions and guidelines

If no rule change is detected, the system does not perform an activation operation.

Insufficient memory can cause rule matching acceleration failures. Unaccelerated rules do not take effect, and rules that have been accelerated are not affected.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPv4 or IPv6 security policy view.

   `security-policy` { **`ip`** | **`ipv6`** }

3. Activate rule matching acceleration.

   `accelerate enhanced enable`

Disabling a security policy rule

1. Enter system view.

   `system-view`

2. Enter IPv4 or IPv6 security policy view.

   `security-policy` { **`ip`** | **`ipv6`** }

3. Enter security policy rule view.

   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }

4. Disable the security policy rule.

   `disable`

   By default, a security policy rule is enabled.

Renaming a security policy rule

1. Enter system view.

   `system-view`

2. Enter IPv4 or IPv6 security policy view.

   `security-policy` { **`ip`** | **`ipv6`** }

3. Rename a security policy rule.

   `rule` **rename** *`old-name new-name`*

Configuring security policy rule groups

Creating a security policy rule group

About this task

Perform this task to create a security policy rule group and add security policy rules to the group.

Restrictions and guidelines

To add a list of security policy rules, make sure the end rule is listed behind the start rule and the specified rules do not belong to any other security policy rule group.

A security policy rule group can contain only IPv4 rules or IPv6 rules.

Procedure

1. Enter system view.
   `system-view`
2. Enter IPv4 or IPv6 security policy view.
   `security-policy` { **`ip`** | **`ipv6`** }
3. Create a security policy rule group and add security policy rules to the group.
   `group name` *`group-name`* [**`from`** *`rule-name1`* **`to`** *`rule-name2`*] [**`description`** *`description-text`*] [**`disable`** | **`enable`**]

Specifying a security policy rule group for a security policy rule

1. Enter system view.
   `system-view`
2. Enter IPv4 or IPv6 security policy view.
   `security-policy` { **`ip`** | **`ipv6`** }
3. Enter security policy rule view.
   `rule` { *`rule-id`* | [*`rule-id`*] **`name`** *`rule-name`* }
4. Specify a security policy rule group for the security policy rule.
   `parent-group` *`group-name`*

Moving a security policy rule group

About this task

Perform this task to move a security policy rule group to change the match order of security policy rules.

Restrictions and guidelines

If you specify a target security policy rule that belongs to a security policy rule group, follow these restrictions and guidelines:

- If the target rule is neither the start nor end rule of the group, you cannot move a security policy rule group to the place before or after the rule.
- If the target rule is the start rule of the group, you can only move a security policy rule group to the place before the rule.

- If the target rule is the end rule of the group, you can only move a security policy rule group to the place after the rule.

You can move a security policy rule group before or after a security policy rule or group of the same type (IPv4 or IPv6).

Procedure

1. Enter system view.
   system-view
2. Enter IPv4 or IPv6 security policy view.
   security-policy { **ip** | **ipv6** }
3. Move a security policy rule group.
   group move *group-name1* { **after** | **before** } { **group** *group-name2* | **rule** *rule-name* }

Renaming a security policy rule group

1. Enter system view.
   system-view
2. Enter IPv4 or IPv6 security policy view.
   security-policy { **ip** | **ipv6** }
3. Rename a security policy rule group.
   group rename *old-name new-name*

Setting the time for fast output of security policy settings as logs

About this task

After the **customlog format security-policy sgcc** command is executed, the device fast outputs settings of enabled security policies as logs in SGCC format every day at the specified time. For more information about fast log output, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.
   system-view
2. Set the time at which the device fast outputs security policy settings as logs every day.
   security-policy config-logging send-time *time*
   By default, the device fast outputs security policy settings as logs every day at 0 o'clock.

Display and maintenance commands for object policies

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display security policy configuration.	**display security-policy** { **ip** | **ipv6** }
Display security policy statistics.	**display security-policy statistics** { **ip** | **ipv6** } [**rule** *rule-name*]
Clear security policy statistics.	**reset security-policy statistics** [**ip** | **ipv6**] [**rule** *rule-name*]

Security policy configuration examples

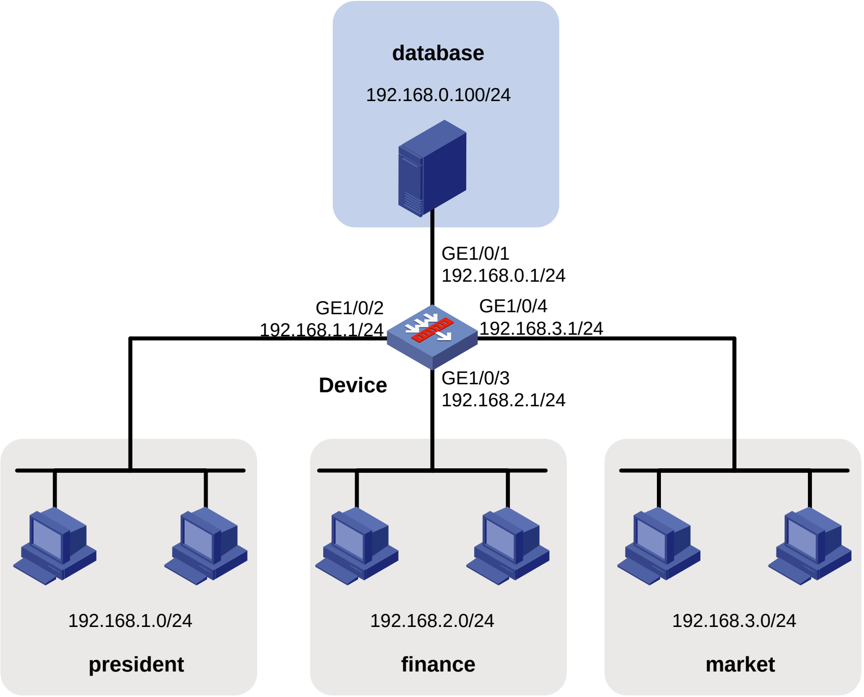
Example: Configuring an IPv4 security policy

Network configuration

Configure security policy to achieve the following goals:

- The president office can access the financial database server through HTTP at any time.
- The financial office can access the financial database server through HTTP from 8:00 to 18:00 on weekdays.
- The marketing office cannot access the financial database server through HTTP at any time.

Figure 6 Network diagram



Procedure

1. Assign IP addresses to interfaces.

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
```

```
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name database
[Device-security-zone-database] import interface gigabitethernet 1/0/1
[Device-security-zone-database] quit
[Device] security-zone name president
[Device-security-zone-president] import interface gigabitethernet 1/0/2
[Device-security-zone-president] quit
[Device] security-zone name finance
[Device-security-zone-finance] import interface gigabitethernet 1/0/3
[Device-security-zone-finance] quit
[Device] security-zone name market
[Device-security-zone-market] import interface gigabitethernet 1/0/4
[Device-security-zone-market] quit
```

3. Create a time range named **work** to cover 8:00 to 18:00 on weekdays.

```
[Device] time-range work 08:00 to 18:00 working-day
```

4. Configure a security policy:

# Configure a rule named **president-database** to allow the president office to access the financial database server through HTTP at any time.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name president-database
[Device-security-policy-ip-0-president-database] source-zone president
[Device-security-policy-ip-0-president-database] destination-zone database
[Device-security-policy-ip-0-president-database] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-0-president-database] destination-ip-subnet 192.168.0.0
24
[Device-security-policy-ip-0-president-database] service http
[Device-security-policy-ip-0-president-database] action pass
[Device-security-policy-ip-0-president-database] quit
```

# Configure a rule named **finance-database** to allow the financial office to access the financial database server through HTTP from 8:00 to 18:00 on weekdays.

```
[Device-security-policy-ip] rule name finance-database
[Device-security-policy-ip-1-finance-database] source-zone finance
[Device-security-policy-ip-1-finance-database] destination-zone database
[Device-security-policy-ip-1-finance-database] source-ip-subnet 192.168.2.0 24
[Device-security-policy-ip-1-finance-database] destination-ip-subnet 192.168.0.0
24
[Device-security-policy-ip-1-finance-database] service-port tcp destination eq 80
[Device-security-policy-ip-1-finance-database] action pass
[Device-security-policy-ip-1-finance-database] time-range work
[Device-security-policy-ip-1-finance-database] quit
```

# Configure a named **market-database** to prohibit the marketing office from accessing the financial database server through HTTP at any time.

```
[Device-security-policy-ip] rule name market-database
[Device-security-policy-ip-2-market-database] source-zone market
[Device-security-policy-ip-2-market-database] destination-zone database
[Device-security-policy-ip-2-market-database] source-ip-subnet 192.168.3.0 24
[Device-security-policy-ip-2-market-database] destination-ip-subnet 192.168.0.0 24
```

```
[Device-security-policy-ip-2-market-database] service http
[Device-security-policy-ip-2-market-database] action drop
[Device-security-policy-ip-2-market-database] quit
```

5. Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Use a PC in each office to access the Web service of the financial database server through the browser. (Details not shown.)

Example: Configuring domain name-based security policy

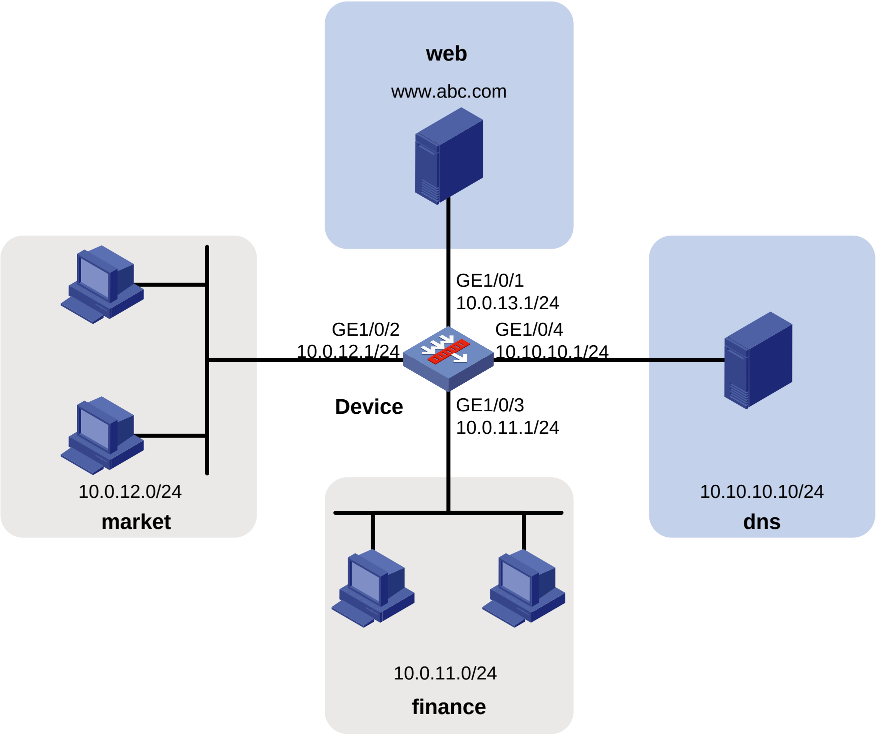
Network configuration

As shown in Figure 7, a Web server with domain name www.abc.com is deployed to process financial affairs. The domain name has been registered on the DNS server.

Configure security policy to achieve the following goals:

- The financial office can access the financial Web server through HTTP.
- The marketing office cannot access the financial Web server through HTTP at any time.

Figure 7 Network diagram



Procedure

1. Assign IP addresses to interfaces.

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.0.13.1 255.255.255.0
```

```
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name web
[Device-security-zone-web] import interface gigabitethernet 1/0/1
[Device-security-zone-web] quit
[Device] security-zone name market
[Device-security-zone-market] import interface gigabitethernet 1/0/2
[Device-security-zone-market] quit
[Device] security-zone name finance
[Device-security-zone-finance] import interface gigabitethernet 1/0/3
[Device-security-zone-finance] quit
[Device] security-zone name dns
[Device-security-zone-dns] import interface gigabitethernet 1/0/4
[Device-security-zone-dns] quit
```

3. Create an IPv4 address object group named **web**. Configure an IPv4 address object with host name **www.abc.com** for the group.

```
[Device] object-group ip address web
[Device-obj-grp-ip-web] network host name www.abc.com
[Device-obj-grp-ip-web] quit
```

4. Specify the IP address of the DNS server as **10.10.10.10**.

```
[Device] dns server 10.10.10.10
```

5. Configure a security policy:

# Configure a rule named **dnslocalout** to allow the device to send packets to the DNS server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name dnslocalout
[Device-security-policy-ip-0-dnslocalout] source-zone local
[Device-security-policy-ip-0-dnslocalout] destination-zone dns
[Device-security-policy-ip-0-dnslocalout] destination-ip-host 10.10.10.10
[Device-security-policy-ip-0-dnslocalout] action pass
[Device-security-policy-ip-0-dnslocalout] quit
```

# Configure a rule named **host-dns** to allow the financial department and marketing department to access the DNS server.

```
[Device-security-policy-ip] rule name host-dns
[Device-security-policy-ip-1-host-dns] source-zone finance
[Device-security-policy-ip-1-host-dns] source-zone market
[Device-security-policy-ip-1-host-dns] destination-zone dns
[Device-security-policy-ip-1-host-dns] source-ip-subnet 10.0.11.0 24
[Device-security-policy-ip-1-host-dns] source-ip-subnet 10.0.12.0 24
[Device-security-policy-ip-1-host-dns] destination-ip-host 10.10.10.10
[Device-security-policy-ip-1-host-dns] service dns-udp
[Device-security-policy-ip-1-host-dns] action pass
[Device-security-policy-ip-1-host-dns] quit
```

# Configure a rule named **finance-web** to allow the financial department to access the financial Web server through HTTP.

```
[Device-security-policy-ip] rule name finance-web
[Device-security-policy-ip-2-finance-web] source-zone finance
[Device-security-policy-ip-2-finance-web] destination-zone web
```

```
[Device-security-policy-ip-2-finance-web] source-ip-subnet 10.0.11.0 24
[Device-security-policy-ip-2-finance-web] destination-ip web
[Device-security-policy-ip-2-finance-web] service http
[Device-security-policy-ip-2-finance-web] action pass
[Device-security-policy-ip-2-finance-web] quit
```

# Configure a rule named **market-web** to forbid the marketing department from accessing the financial Web server through HTTP.

```
[Device-security-policy-ip] rule name market-web
[Device-security-policy-ip-3-market-web] source-zone market
[Device-security-policy-ip-3-market-web] destination-zone web
[Device-security-policy-ip-3-market-web] source-ip-subnet 10.0.12.0 24
[Device-security-policy-ip-3-market-web] destination-ip web
[Device-security-policy-ip-3-market-web] service http
[Device-security-policy-ip-3-market-web] action drop
[Device-security-policy-ip-3-market-web] quit
```

6. Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Use a PC in each office to access the Web service of the financial server. (Details not shown.)

Example: Configuring a security policy based on fuzzy domain name matching

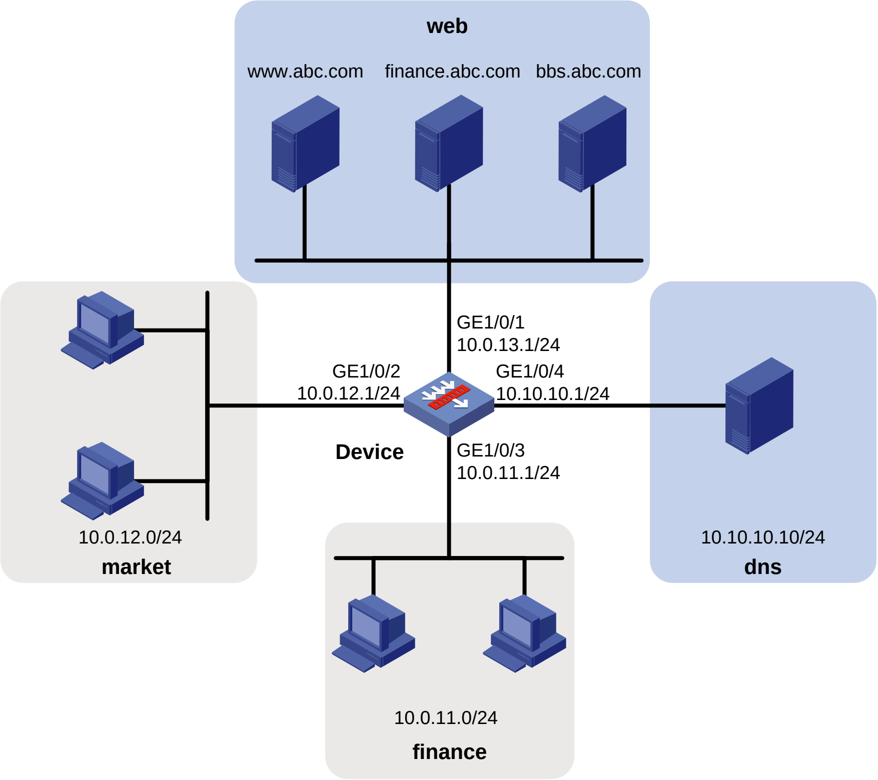
Network configuration

As shown in Figure 8, Web servers with domain names www.abc.com, finance.abc.com, and bbs.abc.com are deployed. The domain names have been registered on the DNS server.

Configure security policy to achieve the following goals:

- The financial office can access all Web servers through HTTP based on fuzzy domain name matching.
- The marketing office can access the bbs.abc.com server through HTTP based on exact domain matching.

Figure 8 Network diagram



Procedure

1. Specify the address of the DNS server for intranet hosts as the IP address of the device.
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.0.13.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name web
   [Device-security-zone-web] import interface gigabitethernet 1/0/1
   [Device-security-zone-web] quit
   [Device] security-zone name market
   [Device-security-zone-market] import interface gigabitethernet 1/0/2
   [Device-security-zone-market] quit
   [Device] security-zone name finance
   [Device-security-zone-finance] import interface gigabitethernet 1/0/3
   [Device-security-zone-finance] quit
   [Device] security-zone name dns
   [Device-security-zone-dns] import interface gigabitethernet 1/0/4
   [Device-security-zone-dns] quit
   ```
4. Create object groups:

# Create an IPv4 address object group named **web**. Configure an IPv4 address object with host name ***.abc.com** for the group. The host name can match all domain names that contain the .abc.com string.

```
[Device] object-group ip address web
[Device-obj-grp-ip-web] network host name *.abc.com
[Device-obj-grp-ip-web] quit
```

# Create an IPv4 address object group named **bbs**. Configure an IPv4 address object with host name **bbs.abc.com** for the group.

```
[Device] object-group ip address bbs
[Device-obj-grp-ip-bbs] network host name bbs.abc.com
[Device-obj-grp-ip-bbs] quit
```

5. Configure DNS settings:

# Enable DNS proxy.

```
[Device] dns proxy enable
```

# Specify the IP address of the DNS server as **10.10.10.10**.

```
[Device] dns server 10.10.10.10
```

6. Configure security policies:

# Configure a rule named **local-dns** to allow the device to send packets to the DNS server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name local-dns
[Device-security-policy-ip-0-local-dns] source-zone local
[Device-security-policy-ip-0-local-dns] destination-zone dns
[Device-security-policy-ip-0-local-dns] destination-ip-host 10.10.10.10
[Device-security-policy-ip-0-local-dns] service dns-udp
[Device-security-policy-ip-0-local-dns] service dns-tcp
[Device-security-policy-ip-0-local-dns] action pass
[Device-security-policy-ip-0-local-dns] quit
```

# Configure a rule named **host-localdns** to allow the financial department and marketing department to access DNS proxy.

```
[Device-security-policy-ip] rule name host-localdns
[Device-security-policy-ip-1-host-localdns] source-zone finance
[Device-security-policy-ip-1-host-localdns] source-zone market
[Device-security-policy-ip-1-host-localdns] destination-zone local
[Device-security-policy-ip-1-host-localdns] source-ip-subnet 10.0.11.0 24
[Device-security-policy-ip-1-host-localdns] source-ip-subnet 10.0.12.0 24
[Device-security-policy-ip-1-host-localdns] service dns-udp
[Device-security-policy-ip-1-host-localdns] service dns-tcp
[Device-security-policy-ip-1-host-localdns] action pass
[Device-security-policy-ip-1-host-dns] quit
```

# Configure a rule named **finance-web** to allow the financial department to access all Web servers through HTTP.

```
[Device-security-policy-ip] rule name finance-web
[Device-security-policy-ip-2-finance-web] source-zone finance
[Device-security-policy-ip-2-finance-web] destination-zone web
[Device-security-policy-ip-2-finance-web] source-ip-subnet 10.0.11.0 24
[Device-security-policy-ip-2-finance-web] destination-ip web
[Device-security-policy-ip-2-finance-web] service http
[Device-security-policy-ip-2-finance-web] action pass
[Device-security-policy-ip-2-finance-web] quit
```

# Configure a rule named **market-web** to allow the marketing department to access the bbs.abc.com server through HTTP.

```
[Device-security-policy-ip] rule name market-web
[Device-security-policy-ip-3-market-web] source-zone market
[Device-security-policy-ip-3-market-web] destination-zone web
[Device-security-policy-ip-3-market-web] source-ip-subnet 10.0.12.0 24
[Device-security-policy-ip-3-market-web] destination-ip bbs
[Device-security-policy-ip-3-market-web] service http
[Device-security-policy-ip-3-market-web] action pass
[Device-security-policy-ip-3-market-web] quit
```

7. Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Use a PC in the financial office to access all Web servers and a PC in the marketing office to access the bbs.abc.com server. (Details not shown.)

Example: Configuring user-based security policy

For more information about user-based security policy configuration examples, see "Configuring user identification."

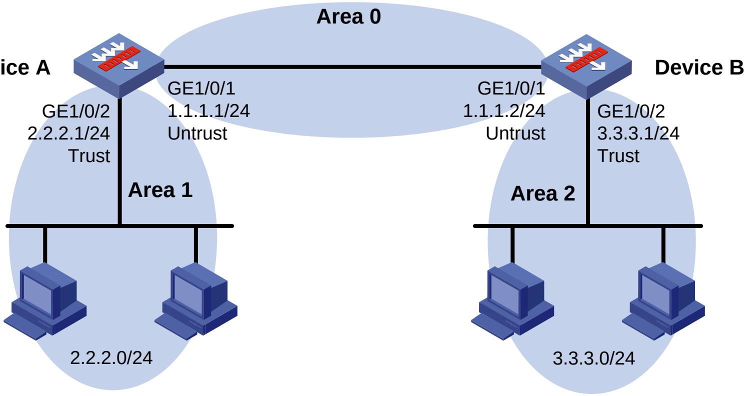
Example: Configuring a security policy for OSPF communication

Network configuration

As shown in Figure 9, OSPF is configured and the network is divided into three areas. Device A and Device B act as ABRs to exchange traffic between areas.

Configure the IPv4 security policy to make sure each router can learn the routes to all network segments in the AS.

Figure 9 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces.

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

3. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can establish OSPF neighbor relations:

   # Configure a rule named **ospflocalin** to permit the packets from security zone **Untrust** to security zone **Local**.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ospflocalin
[DeviceA-security-policy-ip-0-ospflocalin] source-zone untrust
[DeviceA-security-policy-ip-0-ospflocalin] destination-zone local
[DeviceA-security-policy-ip-0-ospflocalin] service ospf
[DeviceA-security-policy-ip-0-ospflocalin] action pass
[DeviceA-security-policy-ip-0-ospflocalin] quit
```

   # Configure a rule named **ospflocalout** to permit the packets from security zone **Local** to security zone **Untrust**.

```
[DeviceA-security-policy-ip] rule name ospflocalout
[DeviceA-security-policy-ip-1-ospflocalout] source-zone local
[DeviceA-security-policy-ip-1-ospflocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ospflocalout] service ospf
[DeviceA-security-policy-ip-1-ospflocalout] action pass
[DeviceA-security-policy-ip-1-ospflocalout] quit
```

   b. Configure rules to permit traffic between the **Untrust** and **Trust** security zones, so devices in area 1 can communicate with devices in area 2.

   # Configure a rule named **trust-untrust** to permit the packets from security zone **Trust** to security zone **Untrust**.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-2-trust-untrust] destination-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-2-trust-untrust] action pass
[DeviceA-security-policy-ip-2-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from security zone **Untrust** to security zone **Trust**.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-3-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-3-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-3-untrust-trust] source-ip-subnet 3.3.3.0 24
```

```
[DeviceA-security-policy-ip-3-untrust-trust] destination-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-3-untrust-trust] action pass
[DeviceA-security-policy-ip-3-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

4. Configure basic OSPF functions.

```
[DeviceA] router id 2.2.2.1
[DeviceA] ospf
[DeviceA-ospf-1] area 0
[DeviceA-ospf-1-area-0.0.0.0] network 1.1.1.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.0] quit
[DeviceA-ospf-1] area 1
[DeviceA-ospf-1-area-0.0.0.1] network 2.2.2.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.1] quit
[DeviceA-ospf-1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] ip address 3.3.3.1 255.255.255.0
[Device-GigabitEthernet1/0/2] quit
```

2. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so **the device can establish OSPF neighbor relations:**

   # Configure a rule named **ospflocalin** to permit the packets from security zone **Untrust** to security zone **Local**.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ospflocalin
[DeviceB-security-policy-ip-0-ospflocalin] source-zone untrust
[DeviceB-security-policy-ip-0-ospflocalin] destination-zone local
[DeviceB-security-policy-ip-0-ospflocalin] service ospf
[DeviceB-security-policy-ip-0-ospflocalin] action pass
[DeviceB-security-policy-ip-0-ospflocalin] quit
```

   # Configure a rule named **ospflocalout** to permit the packets from security zone **Local** to security zone **Untrust**.

```
[DeviceB-security-policy-ip] rule name ospflocalout
[DeviceB-security-policy-ip-1-ospflocalout] source-zone local
[DeviceB-security-policy-ip-1-ospflocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ospflocalout] service ospf
```

```
[DeviceB-security-policy-ip-1-ospflocalout] action pass
[DeviceB-security-policy-ip-1-ospflocalout] quit
```

b. Configure rules to permit traffic between the **Trust** and **Untrust** security zones, so devices in area 1 can communicate with devices in area 2:

# Configure a rule named **trust-untrust** to permit the packets from security zone **Trust** to security zone **Untrust**.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 3.3.3.0 24
[DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 2.2.2.0 24
[DeviceB-security-policy-ip-2-trust-untrust] action pass
[DeviceB-security-policy-ip-2-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from security zone **Untrust** to security zone **Trust**.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-3-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-3-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-3-untrust-trust] source-ip-subnet 2.2.2.0 24
[DeviceB-security-policy-ip-3-untrust-trust] destination-ip-subnet 3.3.3.0 24
[DeviceB-security-policy-ip-3-untrust-trust] action pass
[DeviceB-security-policy-ip-3-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

4. Configure basic OSPF functions.

```
[DeviceB] router id 3.3.3.1
[DeviceB] ospf
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 1.1.1.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] area 2
[DeviceB-ospf-1-area-0.0.0.2] network 3.3.3.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.2] quit
[DeviceB-ospf-1] quit
```

Verifying the configuration

# View detailed information about OSPF neighbors on Device A.

```
[DeviceA] display ospf peer verbose

OSPF Process 1 with Router ID 2.2.2.1
                  Neighbors

 Area 0.0.0.0 interface 1.1.1.1(GigabitEthernet1/0/1)'s neighbors
 Router ID: 3.3.3.1          Address: 1.1.1.2          GR State: Normal
   State: Full  Mode: Nbr is master  Priority: 1
   DR: 1.1.1.1  BDR: 1.1.1.2  MTU: 0
   Options is 0x42 (-|O|-|-|-|-|E|-)
   Dead timer due in 32  sec
   Neighbor is up for 00:07:08
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
   BFD status: Disabled
```

# View OSPF routing information on Device A.

```
[DeviceA] display ospf routing

OSPF Process 1 with Router ID 2.2.2.1
                   Routing Table

 Routing for network
 Destination        Cost     Type    NextHop         AdvRouter       Area
 3.3.3.0/24         2        Inter   1.1.1.2         3.3.3.1         0.0.0.0
 2.2.2.0/24         1        Stub    0.0.0.0         2.2.2.1         0.0.0.1
 1.1.1.0/24         1        Transit 0.0.0.0         2.2.2.1         0.0.0.0

 Total nets: 3
 Intra area: 2  Inter area: 1  ASE: 0  NSSA: 0
```

# Verify that PCs in area 1 can ping PCs in area 2. (Details not shown.)

Contents

Configuring ASPF

About ASPF

Advanced Stateful Packet Filter (ASPF) is proposed to address the issues that a packet-filter firewall cannot solve.

Main functions

An ASPF provides the following main functions:

- **Application layer protocol inspection**—ASPF checks the application layer information of packets, such as the protocol type and port number, and inspects the application layer protocol status for each connection. ASPF maintains the status information of each connection, and based on the status information, determines whether to permit a packet to pass through the firewall into the internal network. In this way, ASPF defends the internal network against attacks.
- **Transport layer protocol inspection**—ASPF checks the transportation layer information of packets. Transportation layer protocol includes TCP, UDP, UDP-Lite, SCTP, Raw IP, ICMP, ICMPv6, and DCCP. For example, ASPF checks a TCP/UDP packet's source and destination addresses and port numbers to determine whether to permit the packet to pass through the firewall into the internal network.
- **ICMP error message dropping**—ASPF inspects the connection information carried in an ICMP error message. If the information does not match the connection, ASPF drops the packet.
- **TCP SYN check**—ASPF checks the first packet of a TCP connection to determine if it is a SYN packet. If it is not a SYN packet, ASPF drops the packet. When a router attached to the network starts up, it can receive a non-SYN packet of an existing TCP connection for the first time. If you do not want to interrupt the existing TCP connection, you can disable the TCP SYN check. The router allows the first non-SYN packet that is used to establish a TCP connection to pass. After the network topology becomes steady, you can enable TCP SYN check again.

ASPF application

At the border of a network, ASPF can work with a packet-filter firewall to provide the network with a more comprehensive security policy that better meets the actual needs. The packet-filter firewall permits or denies packets according to ACL rules. The ASPF records information about the permitted packets to ensure that their return packets can pass through the packet-filter firewall.

Basic ASPF concepts

Single-channel protocol and multichannel protocol

- **Single-channel protocol**—A single-channel protocol establishes only one connection to exchange both control messages and data for a user. SMTP and HTTP are examples of single-channel protocols.
- **Multichannel protocol**—A multichannel protocol establishes more than one connection for a user and transfers control messages and user data through different connections. FTP is one example of multichannel protocols.

Zone pair

A zone pair specifies the source zone and destination zone of a traffic flow to be inspected:

- **Source zone**—A security zone from which the first packet of a traffic flow originates.
- **Destination zone**—A security zone for which the first packet of a traffic flow is destined.

For information about security zones, see "Configuring security zones."

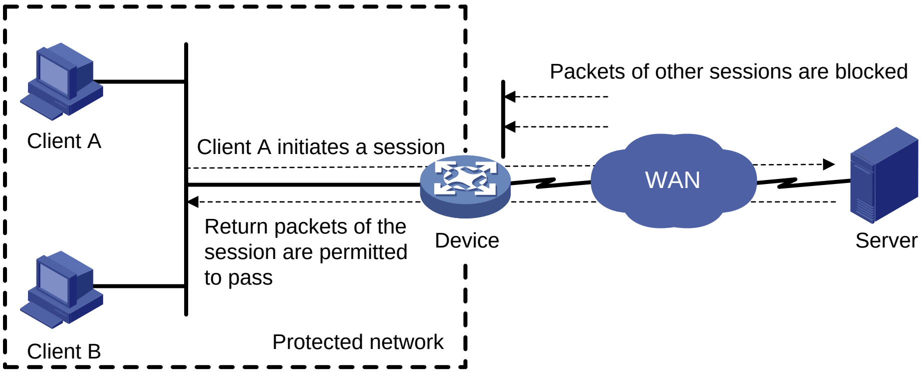
ASPF inspection principles

This section introduces the basic idea of ASPF inspection on application layer and transport layer protocols.

Application layer protocol inspection

As shown in Figure 1, ACLs on the edge device deny incoming packets to the internal network. The ASPF application layer protocol inspection allows return packets from the external network to the internal network.

Figure 1 Application layer protocol inspection



Client A
Client B
Client A initiates a session
Return packets of the session are permitted to pass
Protected network
Device
Packets of other sessions are blocked
WAN
Server

ASPF inspects all application layer sessions as follows:

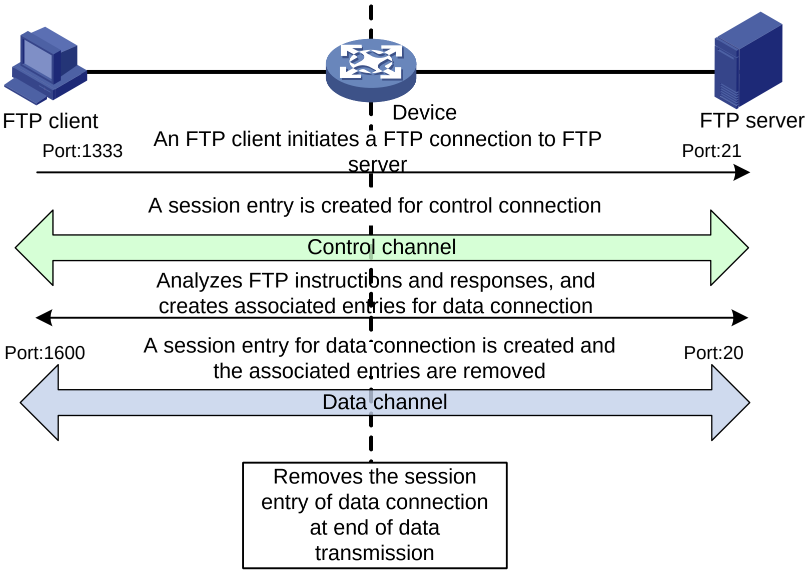
- For a single-channel protocol, such as HTTP, the inspection process is simple.

  ASPF creates a session entry immediately after it detects the session's first packet sent to the external network, and ASPF removes the entry when the connection is terminated.

  The session entry helps record outgoing packets and their return packets. It can maintain the session status and determine whether state transitions of the session are correct. All packets that match a session entry can pass through the packet-filter firewall.
- For a multichannel protocol, ASPF creates session entries, and one or more associated entries to associate the sessions initiated by the same application layer protocol. Associated entries are created during the protocol negotiation and are removed after the negotiation. ASPF uses the associated entries to match the first packets of the sessions. All packets of the sessions matching the associated entries can pass through the packet-filter firewall.

The following uses FTP to explain the process of multichannel application layer protocol inspection.

Figure 2 FTP inspection


FTP client
Device
FTP server
Port:1333
An FTP client initiates a FTP connection to FTP server
Port:21
A session entry is created for control connection
Control channel
Analyzes FTP instructions and responses, and creates associated entries for data connection
Port:1600
A session entry for data connection is created and the associated entries are removed
Port:20
Data channel
Removes the session entry of data connection at end of data transmission

As shown in Figure 2, FTP connections are established and removed as follows:

1. The FTP client initiates an FTP control connection from port 1333 to port 21 of the FTP server.
2. As a result of negotiation, the server initiates a data connection from port 20 to port 1600 of the client.
3. When data transmission times out or ends, the data connection is removed.

ASPF implements FTP inspection during the FTP connection lifetime as follows:

1. ASPF checks the IP packets the FTP client sends to the FTP server to identify TCP-based FTP packets. Based on the port number, ASPF identifies the control connection between the FTP client and server and creates a control connection session entry.
2. ASPF checks each FTP control connection packet, and examines their TCP status based on the control connection session entry. ASPF analyzes the FTP instructions in the control connection packet. If the packet contains a data channel setup instruction, ASPF creates an associated entry for the data connection.
3. For return FTP control connection packets, ASPF examines their TCP status based on the control connection session entry to make packet forwarding decisions.
4. When the FTP data passes through the device, ASPF is triggered to create a session entry for the data connection and remove the associated entry.
5. For returned FTP data packets, ASPF examines their TCP status based on the data connection session entry to make packet forwarding decisions.
6. When the data transmission ends, ASPF removes the data connection session entry. When the FTP connection is removed, ASPF removes the control connection session entry.

Transport layer protocol inspection

The transport layer protocol inspection creates session entries to record the transport layer information of the packets to dynamically filter packets. The transport layer information includes source and destination addresses and port numbers.

The transport layer protocol inspection requires that return packets must match the corresponding packets that are previously sent out of the external interface. The return packets must have the same source/destination addresses and source/destination port numbers as the outgoing packets (but reversed). Otherwise, the return packets are blocked. For multichannel application layer protocols like FTP, the deployment of TCP inspection without application layer inspection leads to failure of establishing a data connection.

Restrictions and guidelines: ASPF configuration

ASPF inspection is required to ensure successful data connections for multichannel protocols when either of the following conditions exists:

- The ALG feature is disabled in other service modules (such as NAT).
- Other service modules with the ALG feature (such as DPI) are not configured.

ASPF inspection is optional for multichannel protocols if ALG is enabled in other service modules or if other service modules with the ALG feature are configured.

Application protocols supported by the **detect** command (except HTTP, SMTP, and TFTP) are multichannel protocols.

ASPF inspection for transport layer protocols is always enabled and is not configurable.

ASPF also supports protocol status validity check for application layer protocols of DNS, FTP, H323, HTTP, SCCP, SIP, and SMTP. ASPF deals with packets with invalid protocol status, depending on the actions you have specified. For other application layer protocols, ASPF does not perform the protocol status validity check, and it only maintains connection status information.

ASPF tasks at a glance

To configure ASPF, perform the following tasks:

1. Configuring an ASPF policy
2. Applying an ASPF policy to a zone pair
3. (Optional.) Sending ICMP error message upon packet dropping by interzone policies applied to zone pairs

Configuring an ASPF policy

1. Enter system view.

   system-view

2. Create an ASPF policy and enter its view.

   aspf-policy *aspf-policy-number*

   After an ASPF policy is created, ASPF inspection for transport layer protocols is always enabled and is not configurable.

3. (Optional.) Configure ASPF inspection for application layer protocols.

   detect { **dns** [**action** { **drop** | **logging** } *] | { **ftp** | **h323** | **http** | **sccp** | **sip** | **smtp** } [**action drop**] | **gtp** | **ils** | **mgcp** | **nbt** | **pptp** | **rsh** | **rtsp** | **sqlnet** | **tftp** | **xdmcp** }

   By default, ASPF inspection for FTP is configured.

   The **action** keyword enables protocol status validity check for application protocols. ASPF takes the predefined actions on packets with invalid protocol status.

4. (Optional.) Enable ICMP error message dropping.

   icmp-error drop

   By default, ICMP error message dropping is disabled. ASPF does not drop faked ICMP error messages.

5. (Optional.) Enable TCP SYN check.

   tcp syn-check

By default, TCP SYN check is disabled. ASPF does not drop the non-SYN packet when it is the first packet to establish a TCP connection.

Applying an ASPF policy to a zone pair

About this task

You can apply an ASPF policy to a zone pair to inspect traffic from the source zone to the destination zone. ASPF compares all packets with session entries. If a packet that is permitted by packet filtering does not match any existing session entries, ASPF creates a new session entry.

ASPF for a zone pair takes effect only when it functions with a packet filter:

- The packet filter allows only solicited access from the source zone to the network that the destination zone connects.
- The ASPF policy compares the packets against session entries and allows matching packets from the source zone to the destination zone. The policy also allows return packets from the destination zone to the source zone.

Procedure

1. Enter system view.

   `system-view`

2. Enter zone pair view.

   `zone-pair security source` *`source-zone-name`* **`destination`** *`destination-zone-name`*

   For information about configuring a zone pair, see "Configuring security zones."

3. Apply an ASPF policy to the zone pair.

   `aspf apply policy` *`aspf-policy-number`*

   By default, the predefined ASPF policy is applied to the zone pair.

   With the predefined policy, ASPF inspects FTP packets and packets of all transport layer protocols, but it does not perform ICMP error message check or TCP SYN packet check.

Sending ICMP error message upon packet dropping by interzone policies applied to zone pairs

About this task

By default, the device drops packets that do not match interzone policies applied to zone pairs, and it does not send ICMP error messages for the dropping events. This mechanism reduces useless packets transmitted over the network and saves bandwidth.

Enable this feature when you use traceroute because ICMP error messages in this situation are required.

Procedure

1. Enter system view.

   `system-view`

2. Enable the device to send ICMP error messages upon packet dropping by interzone policies applied to zone pairs.

   `aspf icmp-error reply`

By default, the device does not send ICMP error messages when the device drops packets that do not match interzone policies applied to zone pairs.

Display and maintenance commands for ASPF

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the configuration of all ASPF policies and their applications to interfaces.	**display aspf all**
Display the configuration of an ASPF policy.	**display aspf policy** { *aspf-policy-number* | **default** }
Display ASPF sessions.	**display aspf session** [**ipv4** | **ipv6**] [**slot** *slot-number*] [**verbose**]
Clear ASPF session statistics.	**reset aspf session** [**ipv4** | **ipv6**] [**slot** *slot-number*]

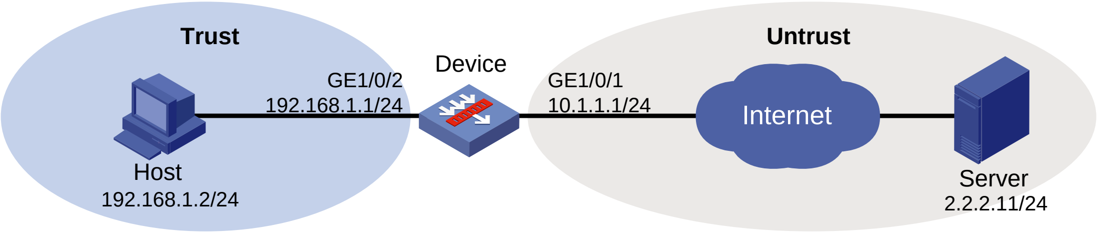
ASPF configuration examples

Example: Configuring ASPF FTP application inspection

Network configuration

Configure an ASPF policy on the device to inspect the FTP traffic flows passing through the device. Only return packets for FTP connections initiated by users on the internal network are permitted to pass through the device and get into the internal network. All other types of packets from the external network to the internal network are blocked.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 10.1.1.2.

```
[Device] ip route-static 2.2.2.0 24 10.1.1.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

4. Configure ACL 3500 to permit the IP packets from the host to the Internet.

```
[Device] acl advanced 3500
[Device-acl-ipv4-adv-3500] rule permit ip source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-adv-3500] quit
```

5. Create ASPF policy 1 for FTP inspection.

```
[Device] aspf policy 1
[Device-aspf-policy-1] detect ftp
[Device-aspf-policy-1] quit
```

6. Create a zone pair with the source security zone **Trust** and destination zone **Untrust**. Apply the ACL to the zone pair to permit the IP packets between the host and the Internet, and apply the ASPF policy to the zone pair.

```
[Device] zone-pair security source trust destination untrust
[Device-zone-pair-security-Trust-Untrust] packet-filter 3500
[Device-zone-pair-security-Trust-Untrust] aspf apply policy 1
[Device-zone-pair-security-Trust-Untrust] quit
```

Verifying the configuration

# Verify that an ASPF session has been established for the FTP connection between the host and the server.

```
<Device> display aspf session ipv4
Slot 0:
Initiator:
  Source      IP/port: 192.168.1.2/1877
  Destination IP/port: 2.2.2.11/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
    Source security zone: Trust
Total sessions found: 1
```

# Verify that only return packets that match the entries can enter the internal network. (Details not shown.)

Contents

Managing sessions

About session management

Session management is a common module, providing basic services for NAT, ASPF, and attack detection and protection to implement their session-based services.

Session management defines packet exchanges at transport layer as sessions. It updates session states and ages out sessions according to data flows from the initiators or responders. Session management allows multiple features to process the same service packet.

Session management operation

Session management tracks the session status by inspecting the transport layer protocol information. It performs unified status maintenance and management of all connections based on session tables and relation tables.

When a connection request passes through the device from a client to a server, the device creates a session entry. The entry can contain the request and response information, such as:

- Source IP address and port number.
- Destination IP address and port number.
- Transport layer protocol.
- Application layer protocol.
- Protocol state of the session.

A multichannel protocol requires that the client and the server negotiate a new connection based on an existing connection to implement an application. Session management enables the device to create a relation entry for each connection during the negotiation phase. The entry is used to associate the connection with the application. Relation entries will be removed after the associated connections are established.

If the destination IP address of a packet is a multicast IP address, the packet will be forwarded out of multiple ports. When a multicast connection request is received on an inbound interface, the device performs the following operations:

- Creates a multicast session entry on the inbound interface.
- Creates a corresponding multicast session entry for each outbound interface.

Unless otherwise stated, "session entry" in this chapter refers to both unicast and multicast session entries.

Session types

When receiving the first packet of a data flow, the device processes the packet and creates a session entry based on the processing result. For subsequent packets of the data flow, the device performs fast forwarding based on the session entry. For more information about fast forwarding, see *Layer 3—IP Services Configuration Guide*.

Sessions are classified into the following types according to the action taken on the packets that match a session entry:

- **Permit session**—The device permits all packets of a permit session. The device generates a permit session entry for a data flow if it permits the first packet of the data flow.

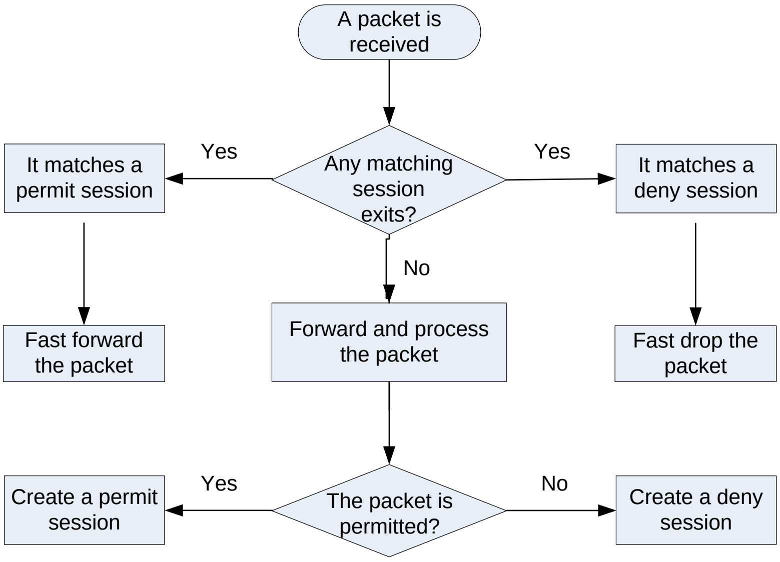
  A permit session can only track connection status. It cannot deny potential attack packets. To deny specific packets, you must use permit sessions together with security features.

- **Deny session**—The device drops all packets of a deny session. The device generates a deny session entry for a data flow if it drops the first packet of the data flow.

Unless otherwise stated, the sessions in this document refer to permit sessions.

Figure 1 shows the workflow of session-based traffic forwarding.

Figure 1 Session-based traffic forwarding



Session management functions

Session management enables the device to provide the following functions:

- Creates sessions for protocol packets, updates session states, and sets aging time for sessions in different protocol states.
- Supports port mapping for application layer protocols (see "Configuring APR"), enabling application layer protocols to use customized ports.
- Sets aging time for sessions based on application layer protocols.
- Supports ICMP/ICMPv6 error packet mapping, enabling the device to search for original sessions according to the payloads in the ICMP/ICMPv6 error packets.

  Because error packets are generated due to host errors, the mapping can help speed up the aging of the original sessions.
- Supports persistent sessions, which are kept alive for a long period of time.
- Supports session management for the control channels and dynamic data channels of application layer protocols, for example, FTP.
- Supports real-time synchronization for sessions and for dynamic entries of session-based services, such as NAT, ALG, and ASPF.

Restrictions and guidelines: Session management configuration

For a TCP session in ESTABLISHED state, the priority order of the associated aging time is as follows:

- Aging time for persistent sessions.

- Aging time for sessions of application layer protocols.
- Aging time for sessions in different protocol states.

If the device has excessive sessions, do not set the aging time shorter than the default for a certain protocol state or an application layer protocol. Short aging time settings can make the device slow in response.

Session management tasks at a glance

To configure session management, perform the following tasks:

- Configure session management timers
  - Setting the session aging time for different protocol states
  - Setting the session aging time for different application layer protocols or applications
  - Specifying persistent sessions
- Specifying the mode for session state machine
- Configuring session synchronization
  - Enabling session synchronization
  - Configuring session dual-active mode
- Configuring session logging
- Configuring session statistics collection
  - Enabling session statistics collection for software fast forwarding
  - Enabling top session statistics
- Configuring deny sessions
- Configuring alarms for abrupt session changes
- Enabling ALG to process fragments

Setting the session aging time for different protocol states

About this task

If a session in a certain protocol state has no packet hit before the aging time expires, the device automatically removes the session.

Procedure

1. Enter system view.

   system-view

2. Set the session aging time for different protocol states.

   session aging-time state { **fin** | **icmp-reply** | **icmp-request** | **icmpv6-reply** | **icmpv6-request** | **rawip-open** | **rawip-ready** | **syn** | **tcp-close** | **tcp-est** | **tcp-time-wait** | **udp-open** | **udp-ready** } *time-value*

   The default aging time for sessions in different protocol states is as follows:

   - FIN_WAIT: 30 seconds.
   - ICMP-REPLY: 30 seconds.
   - ICMP-REQUEST: 60 seconds.
   - ICMPv6-REPLY: 30 seconds.

- ICMPv6-REQUEST: 60 seconds.
- RAWIP-OPEN: 30 seconds.
- RAWIP-READY: 60 seconds.
- TCP SYN-SENT and SYN-RCV: 30 seconds.
- TCP-CLOSE: 2 seconds.
- TCP ESTABLISHED: 3600 seconds.
- TCP-TIME-WAIT: 2 seconds.
- UDP-OPEN: 30 seconds.
- UDP-READY: 60 seconds.

Setting the session aging time for different application layer protocols or applications

About this task

The aging time for sessions of different application layer protocols or applications are valid for TCP sessions in ESTABLISHED state or UDP sessions in READY state. For sessions used by other application layer protocols, the aging time for sessions in different protocol states applies.

Supported application layer protocols or applications specified in this command depend on the APR module. For information about APR, see "Configuring APR."

Procedure

1. Enter system view.

   system-view

2. Set the session aging time for different application layer protocols.

   session aging-time application *application-name time-value*

   By default, the aging time is 1200 seconds for sessions of application layer protocols or applications except for the following sessions:

   - BOOTPC sessions: 120 seconds.
   - BOOTPS sessions: 120 seconds.
   - DNS sessions: 30 seconds.
   - FTP sessions: 3600 seconds.
   - FTP-DATA sessions: 240 seconds.
   - GPRS-DATA sessions: 60 seconds.
   - GPRS-SIG sessions: 60 seconds.
   - GTP-CONTROL sessions: 60 seconds.
   - GTP-USER sessions: 60 seconds.
   - H.225 sessions: 3600 seconds.
   - H.245 sessions: 3600 seconds.
   - HTTPS sessions: 600 seconds.
   - ILS sessions: 3600 seconds.
   - L2TP sessions: 120 seconds.
   - MGCP-CALLAGENT sessions: 60 seconds.
   - MGCP-GATEWAY sessions: 60 seconds.
   - NETBIOS-DGM sessions: 3600 seconds.
   - NETBIOS-NS sessions: 3600 seconds.

- NETBIOS-SSN sessions: 3600 seconds.
- NTP sessions: 120 seconds.
- PPTP sessions: 3600 seconds.
- QQ sessions: 120 seconds.
- RAS sessions: 300 seconds.
- RIP sessions: 120 seconds.
- RSH sessions: 60 seconds.
- RTSP session: 3600 seconds.
- SCCP sessions: 3600 seconds.
- SIP sessions: 300 seconds.
- SNMP sessions: 120 seconds.
- SNMPTRAP sessions: 120 seconds.
- SQLNET sessions: 600 seconds.
- STUN sessions: 600 seconds.
- SYSLOG sessions: 120 seconds.
- TACACS-DS sessions: 120 seconds.
- TFTP sessions: 60 seconds.
- WHO sessions: 120 seconds.
- XDMCP sessions: 3600 seconds.

Specifying persistent sessions

About this task

This task is only for TCP sessions in ESTABLISHED state. You can specify TCP sessions that match the permit statements in the specified ACL as persistent sessions, and set longer lifetime or never-age-out persistent sessions.

A persistent session is not removed until one of the following events occurs:

- The session entry ages out.
- The device receives a connection close request from the initiator or responder.
- You manually clear the session entries.

Procedure

1. Enter system view.
   system-view
2. Specify persistent sessions.
   session persistent acl [**ipv6**] *acl-number* [**aging-time** *time-value*]

Specifying the mode for session state machine

About this task

When asymmetric-path traffic exists in a hot backup system operating in session active/standby mode, set the mode of session state machine to loose to avoid abnormal traffic loss.

When asymmetric-path traffic exists in a hot backup system operating in session dual-active mode, set the mode of session state machine to compact for disconnected sessions to age out timely.

As a best practice, change the mode of session state machine only when asymmetric-path traffic exists. This feature degrades performance of session-based security check. Make sure you are fully aware of its impact when you use it on a live network.

Procedure

1. Enter system view.

   system-view

2. Specify the mode for session state machine.

   session state-machine mode { **compact** | **loose** }

   By default, session state machine is in strict mode.

Enabling session synchronization

About this task

This feature enables devices to synchronize sessions and dynamic entries of session-based services. Typically, these devices back up each other and use a virtual IP address to communicate with a peer device. When the primary device fails, a backup device takes over to process and forward service traffic. The failover process is transparent to the peer device, and does not interrupt ongoing services.

This feature also provides automatic backup service for the device. The system automatically backs up session tables and relation tables that are generated by applications that use ALG. These applications include H.323, SIP, and ILS.

Enable session synchronization for DNS and HTTP in the following situations:

- Users are aware that the current HTTP or DNS sessions will last for a long time.
- HTTP or DNS session backup is required.

In a network that has asymmetric traffic, heavy service traffic might cause service delay or service unavailable because sessions cannot be backed up timely. For example, one device forwards the TCP SYN packets, and another device forwards its ACK packets. If the session tables of the two devices are not synchronized, the TCP packets will be dropped because of state error. To resolve this issue, enable session synchronization for asymmetric traffic.

Restrictions and guidelines

On an IRF fabric, the NAT configuration is dependent of session synchronization. If session synchronization is enabled, make sure NAT is configured on global interfaces such as aggregate interfaces and redundant interfaces. If you configure NAT on physical ports, disable session synchronization as a best practice.

This feature cannot be used together with RBM-based hot backup, which is enabled by using the **hot-backup enable** command. For information about RBM-based hot backup, see *High Availability Command Reference*.

Procedure

1. Enter system view.

   system-view

2. Enable session synchronization.

   session synchronization enable [**asymmetric**]

   By default, session synchronization is disabled.

   After session synchronization is enabled, session synchronization is enabled for all application layer protocols except DNS and HTTP by default.

3. (Optional.) Enable session synchronization for DNS, HTTP, or both.

   session synchronization { **dns** | **http** } *

By default, session synchronization is disabled for DNS and HTTP.

Configuring session dual-active mode

About this task

In a hot backup system operating in session active/standby mode, only one device processes security services. Session dual-active mode increases load capacity of the system by enabling both devices to process security services.

To balance the service load on the devices, you can use one of the following session creation modes:

- **Hash-based session creation**—A session is created on the device to which its first packet is relayed according to the hash result. The device where a session is created might not be the device that receives the traffic. This mode applies if traffic is unevenly distributed among the devices.
- **Local-based session creation**—A session is created on the device where the first packet of the session arrives. This mode applies if traffic is evenly distributed among the devices.

In a hot backup system operating in session dual-active mode, a device cannot identify the direction of packets in a UDP traffic flow due to UDP mechanisms. By default, if the return packets of a session do not match any sessions, the device creates a new session. This results in the following issues:

- If the security control policy permits a UDP traffic flow in one direction, the return packets of the flow are dropped.
- If the security control policy permits a UDP traffic flow in both directions, two sessions are created for the flow. This affects traffic processing of security services.

To resolve these issues, you can enable transparent transmission for UDP packets. This feature allows a device to relay UDP packets that do not match any sessions to the other device in the hot back system. If the UDP packets also do not match any sessions on the other device, a new session is created locally.

In a hot backup system operating in session dual-active mode, a device can identify the direction of packets in a TCP traffic flow. If the device receives the packets of a new TCP traffic flow, it creates a session. If the return packets of an existing TCP traffic flow do not match any sessions, the device relays the packets to the other device.

Restrictions and guidelines

- As a best practice, enable transparent transmission for UDP packets only when asymmetric UDP traffic exists in the hot backup system and sessions cannot be synchronized timely. This feature degrades forwarding performance. Make sure you are fully aware of the impact of this feature when you use it on a live network.
- Transparent transmission for UDP packets takes effect only when local-based session creation is used. If hash-based session creation is used, the devices do not relay UDP packets.
- In dual-active mode, devices support only Layer 3 forwarding. Layer 2 forwarding is not supported.
- In dual-active mode, devices support only the flow-based policy for flow classification. For more information about flow classification policies, see multi-CPU packet distribution in *Layer 3—IP Services Configuration Guide*.
- AFT is not supported in dual-active mode.

Procedure

1. Enter system view.

```
system-view
```

2. Enable session dual-active mode.

session dual-active enable

By default, session dual-active mode is disabled. The device is operating in session active/standby mode.

3. Enable session synchronization.

   See "Enabling session synchronization."

4. Set the session creation mode.

   session dual-active create-mode { **hash** | **local** }

   By default, local-based session creation is used in session dual-active mode.

5. (Optional.) Enable transparent transmission for UDP packets.

   session dual-active transparent udp enable

   By default, transparent transmission for UDP packets is disabled in session dual-active mode.

6. (Optional.) Set the mode of session state machine to compact.

   See "Specifying the mode for session state machine."

Configuring session logging

About this task

Session logs provide information about user access, IP address translation, and network traffic for security auditing. These logs are sent to the log server or the information center.

The device supports time-based or traffic-based logging:

- **Time-based logging**—The device outputs session logs regularly.
- **Traffic-based logging**—The device outputs a session log when the traffic amount of a session reaches a threshold only when the session statistics collection for software fast forwarding feature is enabled. After outputting a session log, the device resets the traffic counter for the session. The traffic-based thresholds can be byte-based and packet-based. If you set both thresholds, the last configuration takes effect.

If you set both time-based and traffic-based logging, the device outputs a session log when whichever is reached. After outputting a session log, the device resets the traffic counter and restarts the interval for the session.

If you enable session logging but do not enable logging for session creation or deletion, the device does not output a session log when a session entry is created or removed.

Restrictions and guidelines

The session logging feature must work with the flow log or fast log output feature to generate session logs. Session logs can be output in flow log or fast log output format. By default, they are output in flow log format. For information about flow log and fast log output, see *Network Management and Monitoring*.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the threshold for time-based session logging.

   session log time-active *time-value*

   By default, no threshold is set for time-based session logging.

3. (Optional.) Set a threshold for traffic-based logging.

   session log { **bytes-active** *bytes-value* | **packets-active** *packets-value* }

   By default, no threshold is set for traffic-based logging.

4. (Optional.) Enable logging for session creation.

   session log flow-begin

   By default, logging for session creation is disabled.

5. (Optional.) Enable logging for session deletion.

   session log flow-end

   By default, logging for session deletion is disabled.

6. Enter interface view.

   interface *interface-type interface-number*

7. Enable session logging.

   session log enable { **ipv4** | **ipv6** } [**acl** *acl-number*] { **inbound** | **outbound** }

   By default, session logging is disabled.

Enabling session statistics collection for software fast forwarding

About this task

This feature enables the device to collect session-based outbound and inbound packets and bytes. You can display session statistics based on different criteria.

- To display statistics per unicast session, use the **display session table** command.
- To display statistics per unicast packet type, use the **display session statistics** command.
- To display statistics per multicast session, use the **display session table multicast** command.
- To display statistics per multicast packet type, use the **display session statistics multicast** command.

Procedure

1. Enter system view.

   system-view

2. Enable session statistics collection for software fast forwarding.

   session statistics enable

   The default setting varies by device model. For more information, see the command reference.

Enabling top session statistics

About this task

This feature collects the number of sessions for session-based services and ranks the sessions by source address and by destination address separately.

Procedure

1. Enter system view.

   system-view

2. Enable the top session statistics feature.

   session top-statistics enable

By default, the top session statistics feature is disabled.

3. (Optional.) Display top session statistics.

   display session top-statistics { **last-1-hour** | **last-24-hours** | **last-30-days** }

Configuring deny sessions

Enabling the deny session feature

About this task

By default, the device generates sessions only for permitted packets. If the device drops the first packet of a data flow, it will forward subsequent packets of the data flow according to the typical forwarding process. To improve forwarding performance, enable the deny session feature. This feature allows the device to generate a deny session for the dropped first packet of each data flow and perform fast packet dropping based on the deny sessions.

The system deletes deny sessions based on the deny session aging time. The deny session aging time is not refreshed when packets match deny sessions.

The maximum ratio of deny sessions to all sessions limits the number of deny sessions. When the ratio of deny session entries reaches this maximum ratio, the device stops generating deny sessions.

Restrictions and guidelines

The device generates deny sessions only for the packets dropped by the ASPF or connection limit module.

The device does not issue deny session entries to chips.

When session synchronization is enabled, the device synchronizes only permit session entries.

Procedure

1. Enter system view.

   system-view

2. Enable the deny session feature for modules.

   session fast-drop { **aspf** | **connection-limit** } * **enable**

   By default, the deny session feature is disabled.

3. Set the aging time for deny sessions.

   session fast-drop aging-time *time-value*

   By default, the aging time for deny sessions is 3 seconds.

4. Set the maximum ratio of deny sessions to all sessions.

   session fast-drop resource-ratio *ratio*

   By default, the maximum ratio of deny sessions to all sessions is 20‰.

Enabling top deny session statistics

About this task

This feature collects the number of deny sessions for session-based services and ranks the deny sessions by source address and by destination address.

To display the top deny session statistics, use the **display session fast-drop top-statistics** command.

Procedure

1. Enter system view.

   system-view

2. Enable the top deny session statistics feature.

   session fast-drop top-statistics enable

   By default, the top deny session statistics feature is disabled.

Configuring alarms for abrupt session changes

Configuring alarms for abrupt session table usage changes

About this task

Perform this task for the device to generate alarms for abrupt increase or drop in the session table usage. With this feature enabled, the system collects the session table usage at an interval of 10 seconds and checks whether the following indicators reach the corresponding alarm thresholds:

- **Session table usage change in percentage**—Obtained by dividing the difference between the session entry counts at the beginning and end of a collection interval by the session entry count at the beginning of the collection interval.
- **Base session table usage in percentage**—Obtained by dividing the session entry count at the beginning of a collection interval by the supported maximum number of session entries.

If both of the following conditions are met in a detection interval, the system generates an alarm for the abrupt change of the session table usage:

- The session table usage change threshold is reached.
- The base session table usage threshold is crossed.

Procedure

1. Enter system view.

   system-view

2. Enable alarms for abrupt session table usage changes.

   session alarm usage-abrupt enable

   By default, alarms are disabled for abrupt session table usage changes.

3. Set the alarm thresholds for abrupt session table usage changes.

   session alarm usage-abrupt threshold *threshold-value* [**base-threshold** *base-value*]

   By default, the session table usage change threshold is 20%, and the base session table usage threshold is 10%.

Configuring alarms for abrupt session creation rate changes

About this task

Perform this task for the device to generate alarms for abrupt increase or drop in the session creation rate. With this feature enabled, the system collects the session creation rate at an interval of 10 seconds and checks whether the following indicators reach the corresponding alarm thresholds:

- **Session creation rate change in percentage**—Obtained by dividing the difference between the session creation rates at the beginning and end of a collection interval by the session creation rate at the beginning of the collection interval.

- **Base session creation rate in percentage**—Obtained by dividing the session creation rate at the beginning of a collection interval by 100000.

If both of the following conditions are met in a detection interval, the system generates an alarm for the abrupt change of the session creation rate:

- The session creation rate change threshold is reached.
- The base session creation rate threshold is crossed.

Procedure

1. Enter system view.

   `system-view`

2. Enable alarms for abrupt session creation rate changes.

   `session alarm rate-abrupt enable`

   By default, alarms are disabled for abrupt session creation rate changes.

3. Set the alarm thresholds for abrupt session creation rate changes.

   session alarm rate-abrupt threshold *threshold-value* [**base-threshold** *base-value*]

   By default, the session creation rate change threshold is 20%, and the base session creation rate threshold is 10%.

Configuring alarms for abrupt session attempt rate changes

About this task

Perform this task for the device to generate alarms for abrupt increase or drop in the session creation attempt rate. With this feature enabled, the system collects the session creation attempt rate at an interval of 10 seconds and checks whether the following indicators reach the corresponding alarm thresholds:

- **Session attempt rate change in percentage**—Obtained by dividing the difference between the session creation attempt rates at the beginning and end of a collection interval by the session creation attempt rate at the beginning of the collection interval.
- **Base session attempt rate in percentage**—Obtained by dividing the session creation attempt rate at the beginning of a collection interval by 100000.

If both of the following conditions are met in a detection interval, the system generates an alarm for the abrupt change of the session creation attempt rate:

- The session attempt rate change threshold is reached.
- The base session attempt rate threshold is crossed.

Procedure

1. Enter system view.

   `system-view`

2. Enable alarms for abrupt session attempt rate changes.

   `session alarm try-rate-abrupt enable`

   By default, alarms are disabled for abrupt session attempt rate changes.

3. Set the alarm thresholds for abrupt session attempt rate changes.

   session alarm try-rate-abrupt threshold *threshold-value* [**base-threshold** *base-value*]

   By default, the session attempt rate change threshold is 20%, and the base session attempt rate threshold is 10%.

Enabling ALG to process fragments

About this task

This task enables ALG to process fragments of specified protocols. In the current software version, ALG can process only SIP fragments.

Procedure

1. Enter system view.

   system-view

2. Enable ALG to process fragments.

   session alg fragment sip

   By default, ALG does not process fragments.

Display and maintenance commands for session management

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the aging time for sessions of different application layer protocols.	**display session aging-time application**
Display the aging time for sessions in different protocol states.	**display session aging-time state**
Display unicast deny session statistics.	**display session fast-drop statistics** [**summary**] [**slot** *slot-number*]
Display IPv4 unicast deny session entries.	**display session fast-drop table ipv4** [**slot** *slot-number*] [**verbose**]
Display IPv6 unicast deny session entries.	**display session fast-drop table ipv6** [**slot** *slot-number*] [**verbose**]
Display top deny session statistics.	**display session fast-drop top-statistics { last-1-hour | last-24-hours | last-30-days }**
Display relation table entries.	**display session relation-table** { **ipv4** | **ipv6** } [**slot** *slot-number*]
Display unicast session statistics.	**display session statistics** [**history-max** | **summary**] [**slot** *slot-number*]
Display IPv4 unicast session statistics.	**display session statistics ipv4** [[**responder**] { **application** *application-name* | **destination-ip** *destination-ip* | **destination-port** *destination-port* | **destination-zone** *destination-zone-name* | **interface** *interface-type interface-number* | **protocol** { **dccp** | **dns** | **ftp** | **gtp** | **h323** | **http** | **icmp** | **ils** | **mgcp** | **nbt** | **pptp** | **raw-ip** | **rsh** | **rtsp** | **sccp** | **sctp** | **sip** | **smtp** | **sqlnet** | **ssh** | **tcp** |

Task	Command
	telnet | **tftp** | **udp** | **udp-lite** | **xdmcp** } | **security-policy-rule** *rule-name* | **source-ip** *source-ip* | **source-port** *source-port* | **source-zone** *source-zone-name* | **state** { **dccp-closereq** | **dccp-closing** | **dccp-open** | **dccp-partopen** | **dccp-request** | **dccp-respond** | **dccp-timewait** | **icmp-reply** | **icmp-request** | **rawip-open** | **rawip-ready** | **sctp-closed** | **sctp-cookie-echoed** | **sctp-cookie-wait** | **sctp-established** | **sctp-shutdown-ack-sent** | **sctp-shutdown-recd** | **sctp-shutdown-sent** | **tcp-close** | **tcp-close-wait** | **tcp-est** | **tcp-fin-wait** | **tcp-last-ack** | **tcp-syn-recv** | **tcp-syn-sent** | **tcp-syn-sent2** | **tcp-time-wait** | **udp-open** | **udp-ready** | **udplite-open** | **udplite-ready** } | **vpn-instance** *vpn-instance-name* } *] [**slot** *slot-number*]
Display IPv6 unicast session statistics.	**display session statistics ipv6** [[**responder**] { **application** *application-name* | **destination-ip** *destination-ip* | **destination-port** *destination-port* | **destination-zone** *destination-zone-name* | **interface** *interface-type interface-number* | **protocol** { **dccp** | **dns** | **ftp** | **gtp** | **h323** | **http** | **icmpv6** | **ils** | **mgcp** | **nbt** | **pptp** | **raw-ip** | **rsh** | **rtsp** | **sccp** | **sctp** | **sip** | **smtp** | **sqlnet** | **ssh** | **tcp** | **telnet** | **tftp** | **udp** | **udp-lite** | **xdmcp** } | **security-policy-rule** *rule-name* | **source-ip** *source-ip* | **source-port** *source-port* | **source-zone** *source-zone-name* | **state** { **dccp-closereq** | **dccp-closing** | **dccp-open** | **dccp-partopen** | **dccp-request** | **dccp-respond** | **dccp-timewait** | **icmpv6-reply** | **icmpv6-request** | **rawip-open** | **rawip-ready** | **sctp-closed** | **sctp-cookie-echoed** | **sctp-cookie-wait** | **sctp-established** | **sctp-shutdown-ack-sent** | **sctp-shutdown-recd** | **sctp-shutdown-sent** | **tcp-close** | **tcp-close-wait** | **tcp-est** | **tcp-fin-wait** | **tcp-last-ack** | **tcp-syn-recv** | **tcp-syn-sent** | **tcp-syn-sent2** | **tcp-time-wait** | **udp-open** | **udp-ready** | **udplite-open** | **udplite-ready** } | **vpn-instance** *vpn-instance-name* } *] [**slot** *slot-number*]
Display multicast session statistics.	**display session statistics multicast** [**slot** *slot-number*]
Display IPv4 unicast session table entries.	**display session table ipv4** [**slot** *slot-number*] [[**responder**] { **application** *application-name* | **destination-ip** *start-destination-ip* [*end-destination-ip*] | **destination-port** *destination-port* | **destination-zone** *destination-zone-name* | **interface** *interface-type interface-number* | **protocol** { **dccp** | **icmp** | **raw-ip** | **sctp** | **tcp** | **udp** |

Task	Command
	udp-lite } | **security-policy-rule** *rule-name* | **source-ip** *start-source-ip* [*end-source-ip*] | **source-port** *source-port* | **source-zone** *source-zone-name* | **state** { **dccp-closereq** | **dccp-closing** | **dccp-open** | **dccp-partopen** | **dccp-request** | **dccp-respond** | **dccp-timewait** | **icmp-reply** | **icmp-request** | **rawip-open** | **rawip-ready** | **sctp-closed** | **sctp-cookie-echoed** | **sctp-cookie-wait** | **sctp-established** | **sctp-shutdown-ack-sent** | **sctp-shutdown-recd** | **sctp-shutdown-sent** | **tcp-close** | **tcp-close-wait** | **tcp-est** | **tcp-fin-wait** | **tcp-last-ack** | **tcp-syn-recv** | **tcp-syn-sent** | **tcp-syn-sent2** | **tcp-time-wait** | **udp-open** | **udp-ready** | **udplite-open** | **udplite-ready** } | **vpn-instance** *vpn-instance-name* } *] [**verbose**]
Display IPv6 unicast session table entries.	**display session table ipv6** [**slot** *slot-number*] [[**responder**] { **application** *application-name* | **destination-ip** *start-destination-ip* [*end-destination-ip*] | **destination-port** *destination-port* | **destination-zone** *destination-zone-name* | **interface** *interface-type interface-number* | **protocol** { **dccp** | **icmpv6** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** } | **security-policy-rule** *rule-name* | **source-ip** *start-source-ip* [*end-source-ip*] | **source-port** *source-port* | **source-zone** *source-zone-name* | **state** { **dccp-closereq** | **dccp-closing** | **dccp-open** | **dccp-partopen** | **dccp-request** | **dccp-respond** | **dccp-timewait** | **icmpv6-reply** | **icmpv6-request** | **rawip-open** | **rawip-ready** | **sctp-closed** | **sctp-cookie-echoed** | **sctp-cookie-wait** | **sctp-established** | **sctp-shutdown-ack-sent** | **sctp-shutdown-recd** | **sctp-shutdown-sent** | **tcp-close** | **tcp-close-wait** | **tcp-est** | **tcp-fin-wait** | **tcp-last-ack** | **tcp-syn-recv** | **tcp-syn-sent** | **tcp-syn-sent2** | **tcp-time-wait** | **udp-open** | **udp-ready** | **udplite-open** | **udplite-ready** } | **vpn-instance** *vpn-instance-name* } *] [**verbose**]
Display IPv4 multicast session table entries.	**display session table multicast ipv4** [**slot** *slot-number*] [[**responder**] { **destination-ip** *start-destination-ip* [*end-destination-ip*] | **destination-port** *destination-port* | **protocol** { **dccp** | **icmp** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** } | **source-ip** *start-source-ip* [*end-source-ip*] | **source-port** *source-port* } *] [**verbose**]
Display IPv6 multicast session table entries.	**display session table multicast ipv6** [**slot** *slot-number*] [[**responder**] { **destination-ip** *start-destination-ip* [*end-destination-ip*] |

Task	Command
	destination-port *destination-port* | **protocol** { **dccp** | **icmpv6** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** } | **source-ip** *start-source-ip* [*end-source-ip*] | **source-port** *source-port* } *] [**verbose**]
Display top session statistics.	**display session top-statistics** { **last-1-hour** | **last-24-hours** | **last-30-days** }
Display statistics about transparently transmitted packets in session dual-active mode.	**display session dual-active transparent statistics** [**slot** *slot-number*]
Clear relation table entries.	**reset session relation-table** [**ipv4** | **ipv6**] [**slot** *slot-number*]
Clear unicast session statistics.	**reset session statistics** [**slot** *slot-number*]
Clear multicast session table entries.	**reset session statistics multicast** [**slot** *slot-number*]
Clear IP unicast session table entries.	**reset session table** [**slot** *slot-number*]
Clear IPv4 unicast session table entries.	**reset session table ipv4** [**slot** *slot-number*] [**source-ip** *source-ip*] [**destination-ip** *destination-ip*] [**protocol** { **dccp** | **icmp** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** }] [**source-port** *source-port*] [**destination-port** *destination-port*] [**vpn-instance** *vpn-instance-name*]
Clear IPv6 unicast session table entries.	**reset session table ipv6** [**slot** *slot-number*] [**source-ip** *source-ip*] [**destination-ip** *destination-ip*] [**protocol** { **dccp** | **icmpv6** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** }] [**source-port** *source-port*] [**destination-port** *destination-port*] [**vpn-instance** *vpn-instance-name*]
Clear IP multicast session table entries.	**reset session table multicast** [**slot** *slot-number*]
Clear IPv4 multicast session table entries.	**reset session table multicast ipv4** [**slot** *slot-number*] [**source-ip** *source-ip*] [**destination-ip** *destination-ip*] [**protocol** { **dccp** | **icmp** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** }] [**source-port** *source-port*] [**destination-port** *destination-port*] [**vpn-instance** *vpn-instance-name*]
Clear IPv6 multicast session table entries.	**reset session table multicast ipv6** [**slot** *slot-number*] [**source-ip** *source-ip*] [**destination-ip** *destination-ip*] [**protocol** { **dccp** | **icmpv6** | **raw-ip** | **sctp** | **tcp** | **udp** | **udp-lite** }] [**source-port** *source-port*] [**destination-port** *destination-port*] [**vpn-instance** *vpn-instance-name*]

Contents

Configuring object groups

About object groups

An object group is a group of objects that can be used by an ACL to identify packets. Object groups are divided into the following types:

- **MAC address object group**—A group of MAC address objects used to match the MAC address in a packet.
- **IPv4 address object group**—A group of IPv4 address objects used to match the IPv4 address in a packet or match the user from whom a packet comes.
- **IPv6 address object group**—A group of IPv6 address objects used to match the IPv6 address in a packet or match the user from whom a packet comes.
- **Service object group**—A group of service objects used to match the upper-layer service in a packet.

Restrictions and guidelines: Object group configuration

You cannot edit an object group if the group is used by a global static NAT rule.

Configuring a MAC address object group

1. Enter system view.

   `system-view`

2. Create a MAC address object group and enter its view.

   `object-group mac-address` *`object-group-name`*

   The system has one default IPv4 address object group named **any**.

3. (Optional.) Configure a description for the MAC address object group.

   `description` *`text`*

   By default, an object group does not have a description.

4. Configure a MAC address object.

   [*`object-id`*] **`mac`** { *`mac-address`* | **`group-object`** *`group-object-name`* }

5. Configure a description for the MAC address object.

   `object` *`object-id`* **`description`** *`text`*

   By default, a MAC address object does not have a description.

Configuring an IPv4 address object group

1. Enter system view.

   `system-view`

2. Create an IPv4 address object group and enter its view.

   `object-group ip address` *`object-group-name`*

   The system has one default IPv4 address object group named **any**.

3. (Optional.) Configure a description for the IPv4 address object group.

   description *text*

   By default, an object group does not have a description.

4. Configure an IPv4 address object.

   [*object-id*] **network** { **host** { **address** *ip-address* | **name** *host-name* [**vpn-instance** *vpn-instance-name*] } | **subnet** *ip-address* { *mask-length* | *mask* | **wildcard** *wildcard* } | **range** *ip-address1 ip-address2* | **group-object** *object-group-name* | **user** *user-name* [**domain** *domain-name*] | **user-group** *user-group-name* [**domain** *domain-name*] }

5. Configure a description for the IPv4 address object.

   object *object-id* **description** *text*

   By default, an IPv4 address object does not have a description.

6. Exclude an IPv4 address or a subnet from the IPv4 address object.

   object-id **network exclude** { *ip-address* | **subnet** *ip-address* { *mask-length* | *mask* } }

   By default, no IPv4 address in an IPv4 address object is excluded.

7. (Optional.) Specify a security zone for the IPv4 address object group.

   security-zone *security-zone-name*

   By default, no security zone is specified for an IPv4 address object group.

Configuring an IPv6 address object group

1. Enter system view.

   system-view

2. Create an IPv6 address object group and enter its view.

   object-group ipv6 address *object-group-name*

   The system has one default IPv6 address object group named **any**.

3. (Optional.) Configure a description for the IPv6 address object group.

   description *text*

   By default, an object group does not have a description.

4. Configure an IPv6 address object.

   [*object-id*] **network** { **host** { **address** *ipv6-address* | **name** *host-name* [**vpn-instance** *vpn-instance-name*] } | **subnet** *ipv6-address prefix-length* | **range** *ipv6-address1 ipv6-address2* | **group-object** *object-group-name* | **user** *user-name* [**domain** *domain-name*] | **user-group** *user-group-name* [**domain** *domain-name*] }

5. Configure a description for an IPv6 address object.

   object *object-id* **description** *text*

   By default, an IPv6 address object does not have a description.

6. Exclude an IPv6 address or a subnet from the IPv6 address object.

   object-id **network exclude** { *ip-address* | **subnet** *ipv6-address prefix-length* }

   By default, no IPv6 address in an IPv6 address object is excluded.

7. (Optional.) Specify a security zone for the IPv6 address object group.

   security-zone *security-zone-name*

   By default, no security zone is specified for an IPv6 address object group.

Configuring a service object group

1. Enter system view.

   system-view

2. Create a service object group and enter its view.

   object-group service *object-group-name*

   The system has multiple default service object groups.

3. (Optional.) Configure a description for the service object group.

   description *text*

   By default, an object group does not have a description.

4. Configure a service object.

   [*object-id*] **service** { *protocol* [{ **source** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } | **destination** { { **eq** | **lt** | **gt** } *port* | **range** *port1 port2* } } * | *icmp-type icmp-code* | *icmpv6-type icmpv6-code*] | **group-object** *object-group-name* }

5. Configure a description for the service object.

   object *object-id* **description** *text*

   By default, a service object does not have a description.

Renaming an object group

1. Enter system view.

   system-view

2. Rename an object group.

   object-group rename *old-object-group-name new-object-group-name*

   You can only rename non-default object groups.

Configuring aging of DNS-resolved IP addresses from host names

About this task

In load balancing scenarios where one host name maps to several IP addresses, DNS-resolved IP address for a host name changes between these mapping addresses. Upon every change, the object group module notifies relevant policies (such as security policy) of the change, which causes policies to submit changes frequently and consumes memory. To resolve this issue, you can enable aging of DNS-resolved IP addresses from host names.

With this feature enabled, the system maintains an IP address group for each host name. If a resolved IP address is not in the group, the system adds the address to the group and notifies relevant policies of the change. If a resolved IP address is in the group, the system does not notify relevant policies.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB680, NFNX3-HDB1080	Yes

NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

Restrictions and guidelines

As a best practice, set the aging time to be longer than the TTL of resolution records on the DNS server.

Procedure

1. Enter system view.

   system-view

2. Enable aging of DNS-resolved IP addresses from host names.

   object-group dns-aging [**time** *aging-time*]

   By default, aging of DNS-resolved IP addresses from host names is disabled.

Display and maintenance commands for object groups

Execute **display** commands in any view.

Task	Command
Display information about object groups.	**display object-group** [{ { **ip** | **ipv6** } **address** | **mac-address** | **service** } [**default**] [**name** *object-group-name*] | **name** *object-group-name*]
Display IPv4 or IPv6 addresses for host names.	**display object-group** { **ip** | **ipv6** } **host** { **object-group-name** *object-group-name* | **name** *host-name* [**vpn-instance** *vpn-instance-name*] } * [**slot** *slot-number* [**cpu** *cpu-number*]]

Contents

Configuring IP source guard

About IPSG

IP source guard (IPSG) prevents spoofing attacks by using an IPSG binding table to filter out illegitimate packets. This feature is typically configured on user-side interfaces.

IPSG operating mechanism

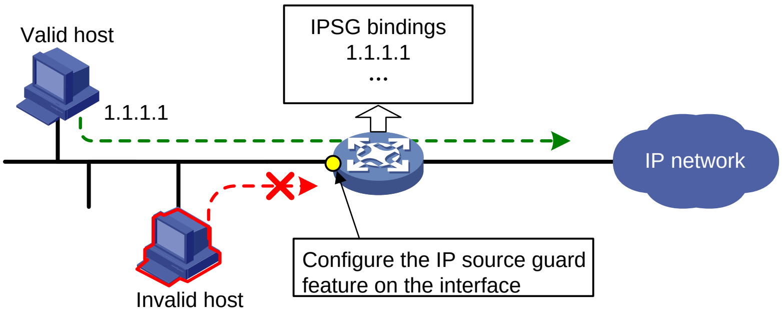
The IPSG binding table contains bindings that bind IP address, MAC address, VLAN, or any combinations. IPSG uses the bindings to match an incoming packet. If a match is found, the packet is forwarded. If no match is found, the packet is discarded.

IPSG is a per-interface packet filter. Configuring this feature on one interface does not affect packet forwarding on another interface.

IPSG bindings can be static or dynamic.

As shown in Figure 1, IPSG forwards only the packets that match an IPSG binding.

Figure 1 IPSG application



Static IPSG bindings

Static IPSG bindings are configured manually. They are suitable for scenarios where few hosts exist on a LAN and their IP addresses are manually configured. For example, you can configure a static IPSG binding on an interface that connects to a server. This binding allows the interface to receive packets only from the server.

Static IPSG bindings on an interface implement the following functions:

- Filter incoming IPv4 or IPv6 packets on the interface.
- Cooperate with ARP attack detection in IPv4 for user validity checking. For information about ARP attack detection, see "Configuring ARP attack protection."

Static IPSG bindings can be global or interface-specific.

- **Global static binding**—Binds the IP address and MAC address in system view. The binding takes effect on all interfaces to filter packets for user spoofing attack prevention.
- **Interface-specific static binding**—Binds the IP address, MAC address, VLAN, or any combination of the items in interface view. The binding takes effect only on the interface to check the validity of users who are attempting to access the interface.

Dynamic IPSG bindings

IPSG automatically obtains user information from other modules to generate dynamic bindings. A dynamic IPSG binding can contain MAC address, IPv4 or IPv6 address, VLAN tag, ingress interface, and binding type. The binding type identifies the source module for the binding, such as DHCP relay.

For example, DHCP-based IPSG bindings are suitable for scenarios where hosts on a LAN obtain IP addresses through DHCP. When a host obtains an IP address through DHCP, each of the following modules generates an entry:

- The DHCP server generates a DHCP binding.
- The DHCP relay agent generates a DHCP relay entry.

IPSG generates a dynamic IPSG binding based on each of the above entries. IPSG allows only packets from the DHCP clients to pass through.

Dynamic IPv4SG

Dynamic bindings generated based on different source modules are for different usages:

Interface types	Source modules	Binding usage
Layer 2 Ethernet interface	ARP snooping	For cooperation with modules (such as the ARP attack detection module) to provide security services.
Layer 3 Ethernet interface Layer 3 aggregate interface VLAN interface Reth interface Reth subinterface	DHCP relay agent	Packet filtering.
	DHCP server	For cooperation with modules (such as the authorized ARP module) to provide security services.

For information about DHCP relay and DHCP server, see *Layer 3—IP Services Configuration Guide*.

Dynamic IPv6SG

Dynamic IPv6SG bindings are generated based on different source modules.

IPSG tasks at a glance

To configure IPv4SG, perform the following tasks:

1. Enabling IPv4SG on an interface
2. (Optional.) Configuring a static IPv4SG binding

To configure IPv6SG, perform the following tasks:

1. Enabling IPv6SG on an interface
2. (Optional.) Configuring a static IPv6SG binding

Configuring the IPv4SG feature

Enabling IPv4SG on an interface

About this task

When you enable IPSG on an interface, the static and dynamic IPSG are both enabled.

- Static IPv4SG uses static bindings configured by using the **ip source binding** command. For more information, see "Configuring a static IPv4SG binding."
- Dynamic IPv4SG generates dynamic bindings from related source modules. IPv4SG uses the bindings to filter incoming IPv4 packets based on the matching criteria specified in the **ip verify source** command.

Restrictions and guidelines

> △ **CAUTION:**
>
> If you only enable IPv4SG on an interface but fail to configure static IPv4SG bindings or dynamic IPv4SG bindings, the interface discards all received packets.

To implement static IPv4SG, configure static IPv4SG bindings.

To implement dynamic IPv4SG, make sure ARP snooping, DHCP relay agent, or DHCP server operates correctly on the network.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
   The following interface types are supported:
   - Layer 2 Ethernet interface.
   - Layer 3 Ethernet interface.
   - Layer 3 Ethernet subinterface.
   - Layer 3 aggregate interface.
   - Layer 3 aggregate subinterface.
   - VLAN interface.
   - Reth interface.
   - Reth subinterface.
3. Enable the IPv4SG feature.
   ip verify source { **ip-address** | **ip-address mac-address** | **mac-address** }
   By default, the IPv4SG feature is disabled on an interface.

Configuring a static IPv4SG binding

About this task

You can configure global static and interface-specific static IPv4SG bindings. Interface-specific static and dynamic bindings take priority over global static bindings. An interface first uses the static and dynamic bindings on the interface to match packets. If no match is found, the interface uses the global bindings.

Restrictions and guidelines

Global static bindings take effect on all interfaces on the device.

To configure a static IPv4SG binding for the ARP attack detection feature, make sure the following conditions are met:

- The **ip-address** *ip-address* option, the **mac-address** *mac-address* option, and the **vlan** *vlan-id* option must be specified.
- ARP attack detection must be enabled for the specified VLAN.

Configuring a global static IPv4SG binding

1. Enter system view.

   system-view

2. Configure a global static IPv4SG binding.

   ip source binding ip-address *ip-address* **mac-address** *mac-address*

Configuring a static IPv4SG binding on an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

   The following interface types are supported:

   - Layer 2 Ethernet interface.
   - Layer 3 Ethernet interface.
   - Layer 3 Ethernet subinterface.
   - Layer 3 aggregate interface.
   - Layer 3 aggregate subinterface.
   - VLAN interface.
   - Reth interface.
   - Reth subinterface.

3. Configure a static IPv4SG binding.

   ip source binding { **ip-address** *ip-address* | **ip-address** *ip-address* **mac-address** *mac-address* | **mac-address** *mac-address* } [**vlan** *vlan-id*]

   You can configure the same static IPv4SG binding on different interfaces.

Configuring the IPv6SG feature

Enabling IPv6SG on an interface

About this task

When you enable IPv6SG on an interface, the static and dynamic IPv6SG are both enabled.

- Static IPv6SG uses static bindings configured by using the **ipv6 source binding** command. For more information, see "Configuring a static IPv6SG binding."
- Dynamic IPv6SG generates dynamic bindings from related source modules. IPv6SG uses the bindings to filter incoming IPv6 packets based on the matching criteria specified in the **ipv6 verify source** command.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

   The following interface types are supported:

   - Layer 2 Ethernet interface.
   - Layer 3 Ethernet interface.

- Layer 3 Ethernet subinterface.
- Layer 3 aggregate interface.
- Layer 3 aggregate subinterface.
- VLAN interface.
- Reth interface.
- Reth subinterface.

3. Enable the IPv6SG feature.

```
ipv6 verify source { ip-address | ip-address mac-address |
mac-address }
```

By default, the IPv6SG feature is disabled on an interface.

Configuring a static IPv6SG binding

About this task

You can configure global static and interface-specific static IPv6SG bindings. Interface-specific static and dynamic bindings take priority over global static bindings. An interface first uses the static and dynamic bindings on the interface to match packets. If no match is found, the interface uses the global bindings.

Restrictions and guidelines

Global static bindings take effect on all interfaces on the device.

Configuring a global static IPv6SG binding

1. Enter system view.

```
system-view
```

2. Configure a global static IPv6SG binding.

```
ipv6 source binding ip-address ipv6-address mac-address mac-address
```

Configuring a static IPv6SG binding on an interface

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

The following interface types are supported:

- Layer 2 Ethernet interface.
- Layer 3 Ethernet interface.
- Layer 3 Ethernet subinterface.
- Layer 3 aggregate interface.
- Layer 3 aggregate subinterface.
- VLAN interface.
- Reth interface.
- Reth subinterface.

3. Configure a static IPv6SG binding.

```
ipv6 source binding { ip-address ipv6-address | ip-address ipv6-address
mac-address mac-address | mac-address mac-address } [ vlan vlan-id ]
```

You can configure the same static IPv6SG binding on different interfaces.

Display and maintenance commands for IPSG

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPv4SG bindings.	**display ip source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**dhcp-relay** | **dhcp-server** | **ip-mac-vlan** | **ip-mac-vpn**]] [**ip-address** *ip-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPv6SG bindings.	**display ipv6 source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**ipv6-mac-vlan** | **ipv6-mac-vpn**]] [**ip-address** *ipv6-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] [**slot** *slot-number*]

Contents

Configuring AAA

About AAA

AAA implementation

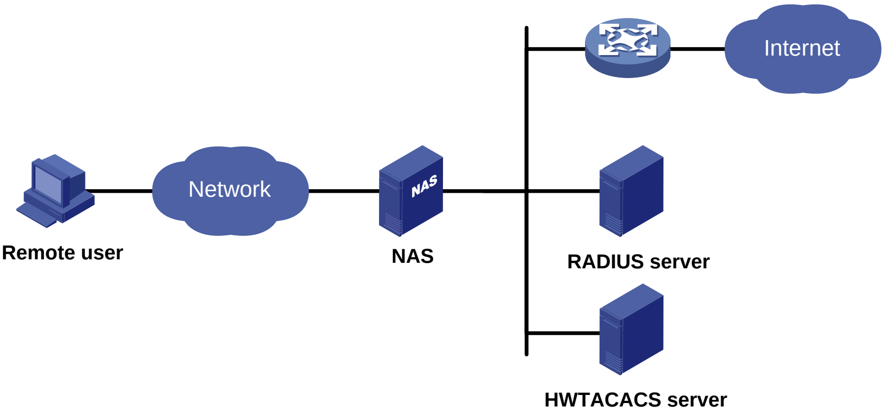
Authentication, Authorization, and Accounting (AAA) provides a uniform framework for implementing network access management. This feature specifies the following security functions:

- **Authentication**—Identifies users and verifies their validity.
- **Authorization**—Grants different users different rights, and controls the users' access to resources and services. For example, you can permit office users to read and print files and prevent guests from accessing files on the device.
- **Accounting**—Records network usage details of users, including the service type, start time, and traffic. This function enables time-based and traffic-based charging and user behavior auditing.

AAA network diagram

AAA uses a client/server model. The client runs on the access device, or the network access server (NAS), which authenticates user identities and controls user access. The server maintains user information centrally. See Figure 1.

Figure 1 AAA network diagram



To access networks or resources beyond the NAS, a user sends its identity information to the NAS. The NAS transparently passes the user information to AAA servers and waits for the authentication, authorization, and accounting result. Based on the result, the NAS determines whether to permit or deny the access request.

AAA has various implementations, including HWTACACS, LDAP, and RADIUS. RADIUS is most often used.

You can use different servers to implement different security functions. For example, you can use an HWTACACS server for authentication and authorization, and use a RADIUS server for accounting.

You can choose the security functions provided by AAA as needed. For example, if your company wants employees to be authenticated before they access specific resources, you would deploy an authentication server. If network usage information is needed, you would also configure an accounting server.

The device performs dynamic password authentication.

RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a distributed information interaction protocol that uses a client/server model. The protocol can protect networks against unauthorized access and is often used in network environments that require both high security and remote user access.

The RADIUS authorization process is combined with the RADIUS authentication process, and user authorization information is piggybacked in authentication responses. RADIUS uses UDP port 1812 for authentication and UDP port 1813 for accounting.

RADIUS was originally designed for dial-in user access, and has been extended to support additional access methods, such as Ethernet and ADSL.

Client/server model

The RADIUS client runs on the NASs located throughout the network. It passes user information to RADIUS servers and acts on the responses to, for example, reject or accept user access requests.

The RADIUS server runs on the computer or workstation at the network center and maintains information related to user authentication and network service access.

The RADIUS server operates using the following process:

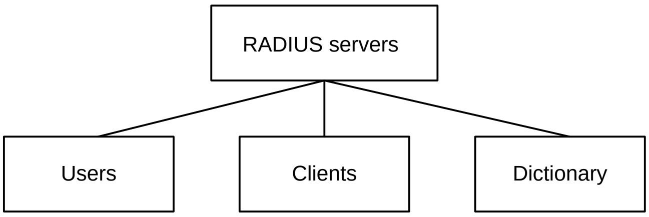
1. Receives authentication, authorization, and accounting requests from RADIUS clients.
2. Performs user authentication, authorization, or accounting.
3. Returns user access control information (for example, rejecting or accepting the user access request) to the clients.

The RADIUS server can also act as the client of another RADIUS server to provide authentication proxy services.

The RADIUS server maintains the following databases:

- **Users**—Stores user information, such as the usernames, passwords, applied protocols, and IP addresses.
- **Clients**—Stores information about RADIUS clients, such as shared keys and IP addresses.
- **Dictionary**—Stores RADIUS protocol attributes and their values.

Figure 2 RADIUS server databases



Information exchange security mechanism

The RADIUS client and server exchange information between them with the help of shared keys, which are preconfigured on the client and server. A RADIUS packet has a 16-byte field called Authenticator. This field includes a signature generated by using the MD5 algorithm, the shared key, and some other information. The receiver of the packet verifies the signature and accepts the packet only when the signature is correct. This mechanism ensures the security of information exchanged between the RADIUS client and server.

The shared keys are also used to encrypt user passwords that are included in RADIUS packets.

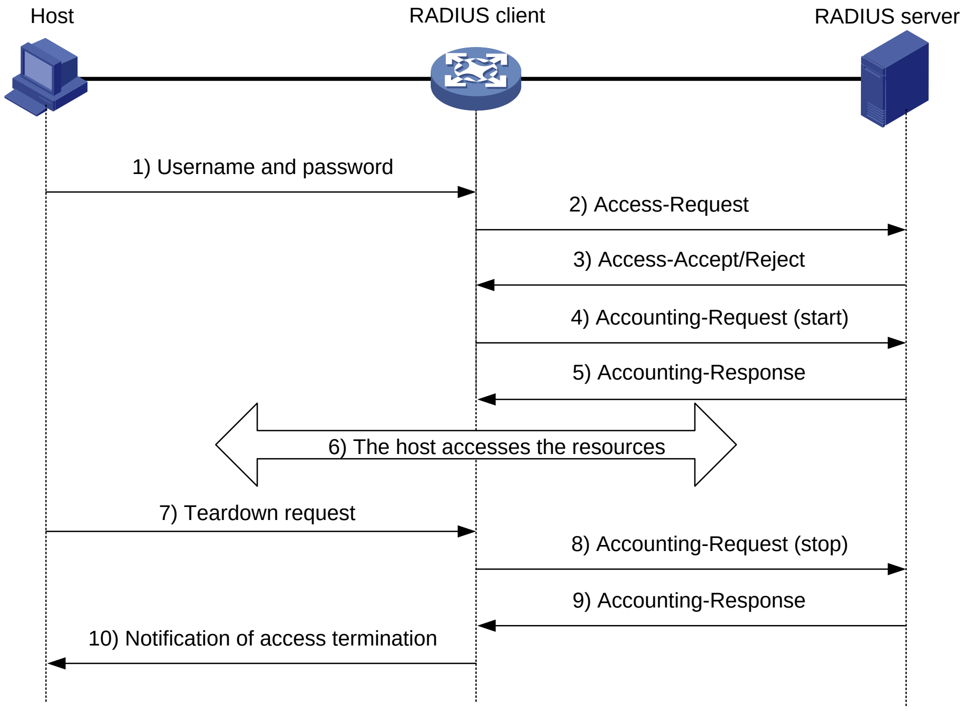
User authentication methods

The RADIUS server supports multiple user authentication methods, such as PAP, CHAP, and EAP.

Basic RADIUS packet exchange process

Figure 3 illustrates the interactions between a user host, the RADIUS client, and the RADIUS server.

Figure 3 Basic RADIUS packet exchange process



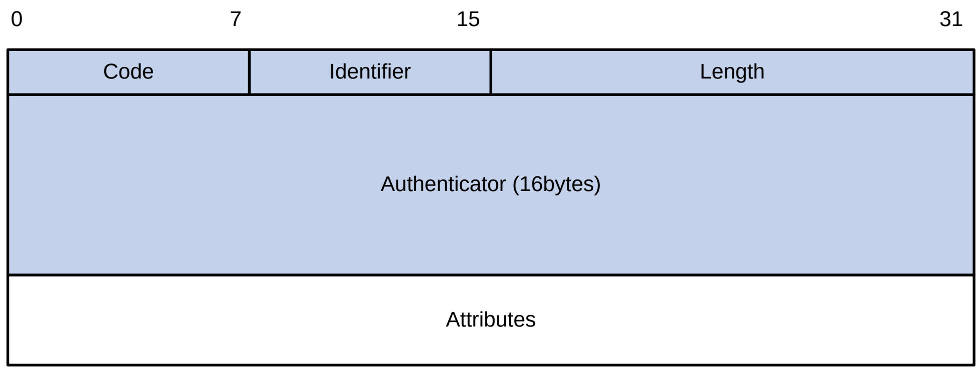
RADIUS uses in the following workflow:

1. The host sends a connection request that includes the user's username and password to the RADIUS client.
2. The RADIUS client sends an authentication request (Access-Request) to the RADIUS server. The request includes the user's password, which has been processed by the MD5 algorithm and shared key.
3. The RADIUS server authenticates the username and password. If the authentication succeeds, the server sends back an Access-Accept packet that contains the user's authorization information. If the authentication fails, the server returns an Access-Reject packet.
4. The RADIUS client permits or denies the user according to the authentication result. If the result permits the user, the RADIUS client sends a start-accounting request (Accounting-Request) packet to the RADIUS server.
5. The RADIUS server returns an acknowledgment (Accounting-Response) packet and starts accounting.
6. The user accesses the network resources.
7. The host requests the RADIUS client to tear down the connection.
8. The RADIUS client sends a stop-accounting request (Accounting-Request) packet to the RADIUS server.
9. The RADIUS server returns an acknowledgment (Accounting-Response) and stops accounting for the user.
10. The RADIUS client notifies the user of the termination.

RADIUS packet format

RADIUS uses UDP to transmit packets. The protocol also uses a series of mechanisms to ensure smooth packet exchange between the RADIUS server and the client. These mechanisms include the timer mechanism, the retransmission mechanism, and the backup server mechanism.

Figure 4 RADIUS packet format



Descriptions of the fields are as follows:

- The Code field (1 byte long) indicates the type of the RADIUS packet. Table 1 gives the main values and their meanings.

  Table 1 Main values of the Code field

Code	Packet type	Description
1	Access-Request	From the client to the server. A packet of this type includes user information for the server to authenticate the user. It must contain the User-Name attribute and can optionally contain the attributes of NAS-IP-Address, User-Password, and NAS-Port.
2	Access-Accept	From the server to the client. If all attribute values included in the Access-Request are acceptable, the authentication succeeds, and the server sends an Access-Accept response.
3	Access-Reject	From the server to the client. If any attribute value included in the Access-Request is unacceptable, the authentication fails, and the server sends an Access-Reject response.
4	Accounting-Request	From the client to the server. A packet of this type includes user information for the server to start or stop accounting for the user. The Acct-Status-Type attribute in the packet indicates whether to start or stop accounting.
5	Accounting-Response	From the server to the client. The server sends a packet of this type to notify the client that it has received the Accounting-Request and has successfully recorded the accounting information.

- The Identifier field (1 byte long) is used to match response packets with request packets and to detect duplicate request packets. The request and response packets of the same exchange process for the same purpose (such as authentication or accounting) have the same identifier.
- The Length field (2 bytes long) indicates the length of the entire packet (in bytes), including the Code, Identifier, Length, Authenticator, and Attributes fields. Bytes beyond this length are considered padding and are ignored by the receiver. If the length of a received packet is less than this length, the packet is dropped.
- The Authenticator field (16 bytes long) is used to authenticate responses from the RADIUS server and to encrypt user passwords. There are two types of authenticators: request authenticator and response authenticator.

- The Attributes field (variable in length) includes authentication, authorization, and accounting information. This field can contain multiple attributes, each with the following subfields:
    - **Type**—Type of the attribute.
    - **Length**—Length of the attribute in bytes, including the Type, Length, and Value subfields.
    - **Value**—Value of the attribute. Its format and content depend on the Type subfield.

Extended RADIUS attributes

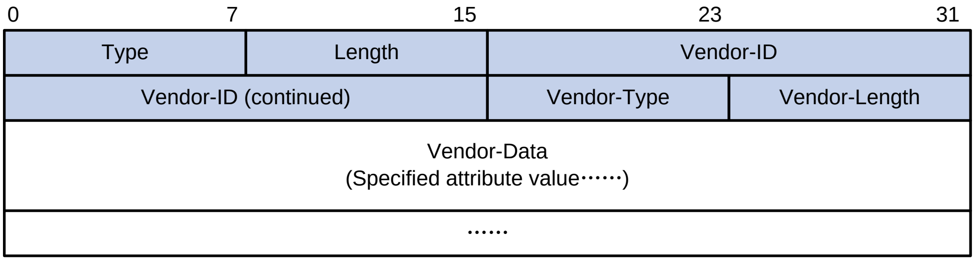
The RADIUS protocol features excellent extensibility. The Vendor-Specific attribute (attribute 26) allows a vendor to define extended attributes. The extended attributes can implement functions that the standard RADIUS protocol does not provide.

A vendor can encapsulate multiple subattributes in the TLV format in attribute 26 to provide extended functions. As shown in Figure 5, a subattribute encapsulated in attribute 26 consists of the following parts:

- **Vendor-ID**—ID of the vendor. The most significant byte is 0. The other three bytes contains a code compliant to RFC 1700.
- **Vendor-Type**—Type of the subattribute.
- **Vendor-Length**—Length of the subattribute.
- **Vendor-Data**—Contents of the subattribute.

The device supports private RADIUS subattributes with a vendor ID of 25506. For more information, see "Appendix C RADIUS subattributes (vendor ID 25506)."

Figure 5 Format of attribute 26



HWTACACS

HW Terminal Access Controller Access Control System (HWTACACS) is an enhanced security protocol based on TACACS (RFC 1492). HWTACACS is similar to RADIUS, and uses a client/server model for information exchange between the NAS and the HWTACACS server.

HWTACACS typically provides AAA services for VPDN, PPP, and terminal users. In a typical HWTACACS scenario, terminal users need to log in to the NAS. Working as the HWTACACS client, the NAS sends users' usernames and passwords to the HWTACACS server for authentication. After passing authentication and obtaining authorized rights, a user logs in to the device and performs operations. The HWTACACS server records the operations that each user performs.

Differences between HWTACACS and RADIUS

HWTACACS and RADIUS have many features in common, such as using a client/server model, using shared keys for data encryption, and providing flexibility and scalability. Table 2 lists the primary differences between HWTACACS and RADIUS.

Table 2 Primary differences between HWTACACS and RADIUS

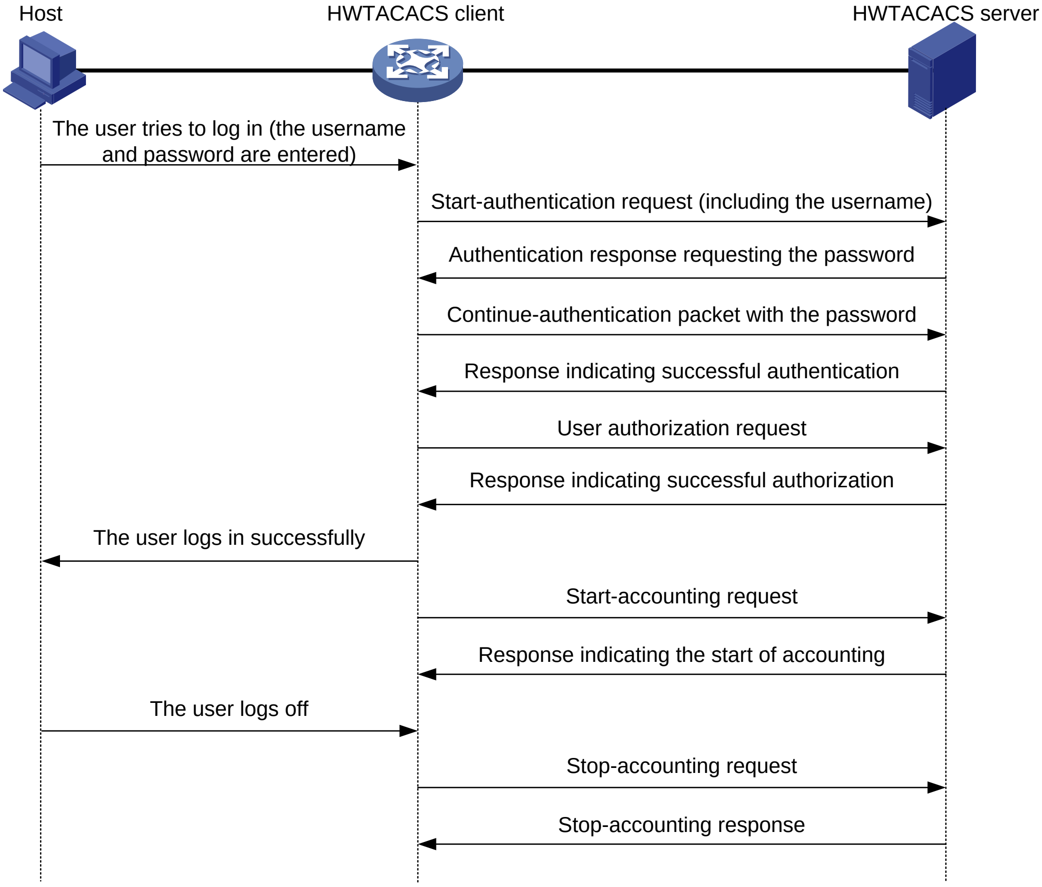
HWTACACS	RADIUS
Uses TCP, which provides reliable network	Uses UDP, which provides high transport efficiency.

HWTACACS	RADIUS
transmission.	
Encrypts the entire packet except for the HWTACACS header.	Encrypts only the user password field in an authentication packet.
Protocol packets are complicated and authorization is independent of authentication. Authentication and authorization can be deployed on different HWTACACS servers.	Protocol packets are simple and the authorization process is combined with the authentication process.
Supports authorization of configuration commands. Access to commands depends on both the user's roles and authorization. A user can use only commands that are permitted by the user roles and authorized by the HWTACACS server.	Does not support authorization of configuration commands. Access to commands solely depends on the user's roles. For more information about user roles, see *Fundamentals Configuration Guide*.

Basic HWTACACS packet exchange process

Figure 6 describes how HWTACACS performs user authentication, authorization, and accounting for a Telnet user.

Figure 6 Basic HWTACACS packet exchange process for a Telnet user


Host
HWTACACS client
HWTACACS server
The user tries to log in (the username and password are entered)
Start-authentication request (including the username)
Authentication response requesting the password
Continue-authentication packet with the password
Response indicating successful authentication
User authorization request
Response indicating successful authorization
The user logs in successfully
Start-accounting request
Response indicating the start of accounting
The user logs off
Stop-accounting request
Stop-accounting response

HWTACACS operates using in the following workflow:

1. A Telnet user requests to Telnet to the device (HWTACACS client) and provides the username and password as instructed by the system.
2. The HWTACACS client sends a start-authentication request that includes the username to the HWTACACS server when it receives the Telnet request.

3. The HWTACACS server sends back an authentication response to request the login password.
4. Upon receipt of the response, the HWTACACS client sends the HWTACACS server a continue-authentication packet that includes the login password.
5. If the authentication succeeds, the HWTACACS server sends back an authentication response to indicate that the user has passed authentication.
6. The HWTACACS client sends a user authorization request packet to the HWTACACS server.
7. If the authorization succeeds, the HWTACACS server sends back an authorization response, indicating that the user is now authorized.
8. Knowing that the user is now authorized, the HWTACACS client pushes its CLI to the user and permits the user to log in.
9. The HWTACACS client sends a start-accounting request to the HWTACACS server.
10. The HWTACACS server sends back an accounting response, indicating that it has received the start-accounting request.
11. The user logs off.
12. The HWTACACS client sends a stop-accounting request to the HWTACACS server.
13. The HWTACACS server sends back a stop-accounting response, indicating that the stop-accounting request has been received.

LDAP

The Lightweight Directory Access Protocol (LDAP) provides standard multiplatform directory service. LDAP was developed on the basis of the X.500 protocol. It improves the following functions of X.500:

- Read/write interactive access.
- Browse.
- Search.

LDAP is suitable for storing data that does not often change. The protocol is used to store user information. For example, LDAP server software Active Directory Server is used in Microsoft Windows operating systems. The software stores the user information and user group information for user login authentication and authorization.

LDAP directory service

LDAP uses directories to maintain the organization information, personnel information, and resource information. The directories are organized in a tree structure and include entries. An entry is a set of attributes with distinguished names (DNs). The attributes are used to store information such as usernames, passwords, emails, computer names, and phone numbers.

LDAP uses a client/server model, and all directory information is stored in the LDAP server. Commonly used LDAP server products include Microsoft Active Directory Server, IBM Tivoli Directory Server, and Sun ONE Directory Server.

LDAP authentication and authorization

AAA can use LDAP to provide authentication and authorization services for users. LDAP defines a set of operations to implement its functions. The main operations for authentication and authorization are the bind operation and search operation.

- The bind operation allows an LDAP client to perform the following operations:
    - Establish a connection with the LDAP server.
    - Obtain the access rights to the LDAP server.
    - Check the validity of user information.
- The search operation constructs search conditions and obtains the directory resource information of the LDAP server.

In LDAP authentication, the client completes the following tasks:

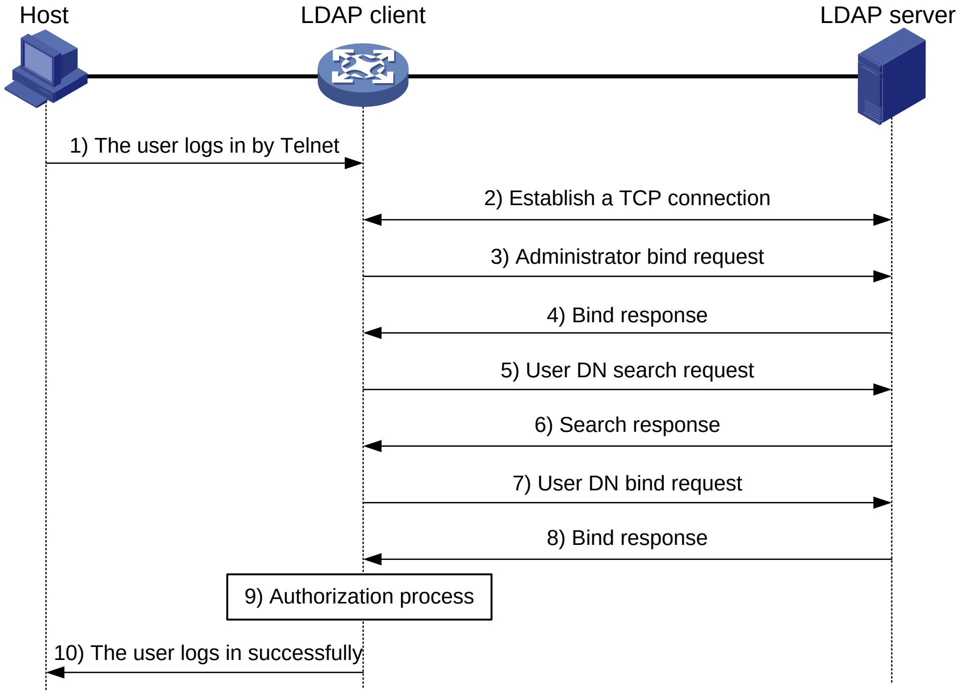
1. Uses the LDAP server administrator DN to bind with the LDAP server. After the binding is created, the client establishes a connection to the server and obtains the right to search.
2. Constructs search conditions by using the username in the authentication information of a user. The specified root directory of the server is searched and a user DN list is generated.
3. Binds with the LDAP server by using each user DN and password. If a binding is created, the user is considered legal.

In LDAP authorization, the client performs the same tasks as in LDAP authentication. When the client constructs search conditions, it obtains both authorization information and the user DN list.

Basic LDAP authentication process

The following example illustrates the basic LDAP authentication process for a Telnet user.

Figure 7 Basic LDAP authentication process for a Telnet user


Host
LDAP client
LDAP server
1) The user logs in by Telnet
2) Establish a TCP connection
3) Administrator bind request
4) Bind response
5) User DN search request
6) Search response
7) User DN bind request
8) Bind response
9) Authorization process
10) The user logs in successfully

The following shows the basic LDAP authentication process:

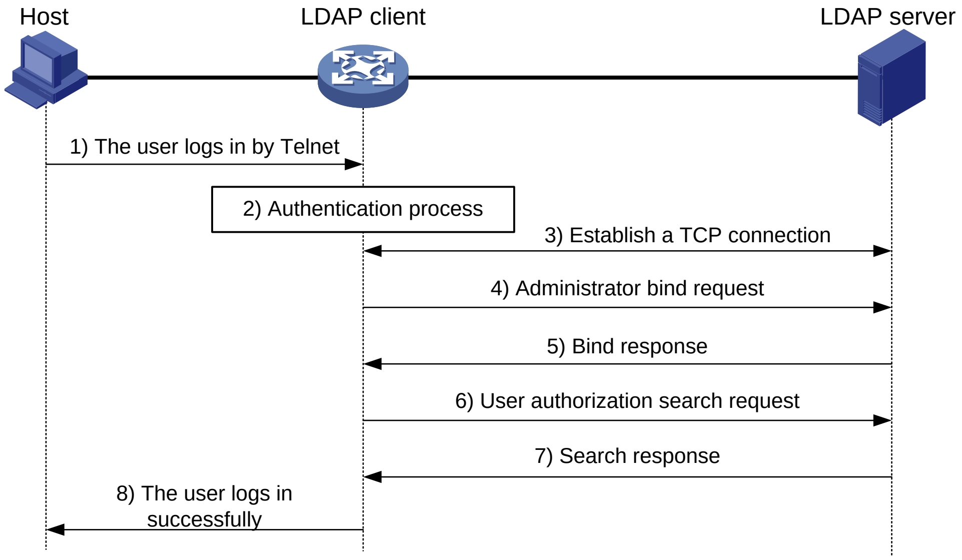
1. A Telnet user initiates a connection request and sends the username and password to the LDAP client.
2. After receiving the request, the LDAP client establishes a TCP connection with the LDAP server.
3. To obtain the right to search, the LDAP client uses the administrator DN and password to send an administrator bind request to the LDAP server.
4. The LDAP server processes the request. If the bind operation is successful, the LDAP server sends an acknowledgment to the LDAP client.
5. The LDAP client sends a user DN search request with the username of the Telnet user to the LDAP server.
6. After receiving the request, the LDAP server searches for the user DN by the base DN, search scope, and filtering conditions. If a match is found, the LDAP server sends a response to notify the LDAP client of the successful search. There might be one or more user DNs found.
7. The LDAP client uses the obtained user DN and the entered user password as parameters to send a user DN bind request to the LDAP server. The server will check whether the user password is correct.

8. The LDAP server processes the request, and sends a response to notify the LDAP client of the bind operation result. If the bind operation fails, the LDAP client uses another obtained user DN as the parameter to send a user DN bind request to the LDAP server. This process continues until a DN is bound successfully or all DNs fail to be bound. If all user DNs fail to be bound, the LDAP client notifies the user of the login failure and denies the user's access request.
9. The LDAP client saves the user DN that has been bound and exchanges authorization packets with the authorization server.
   - If LDAP authorization is used, see the authorization process shown in Figure 8.
   - If another method is expected for authorization, the authorization process of that method applies.
10. After successful authorization, the LDAP client notifies the user of the successful login.

Basic LDAP authorization process

The following example illustrates the basic LDAP authorization process for a Telnet user.

Figure 8 Basic LDAP authorization process for a Telnet user


Host
LDAP client
LDAP server
1) The user logs in by Telnet
2) Authentication process
3) Establish a TCP connection
4) Administrator bind request
5) Bind response
6) User authorization search request
7) Search response
8) The user logs in successfully

The following shows the basic LDAP authorization process:

1. A Telnet user initiates a connection request and sends the username and password to the device. The device will act as the LDAP client during authorization.
2. After receiving the request, the device exchanges authentication packets with the authentication server for the user:
   - If LDAP authentication is used, see the authentication process shown in Figure 7.
     - If the device (the LDAP client) uses the same LDAP server for authentication and authorization, skip to step 6.
     - If the device (the LDAP client) uses different LDAP servers for authentication and authorization, skip to step 4.
   - If another authentication method is used, the authentication process of that method applies. The device acts as the LDAP client. Skip to step 3.
3. The LDAP client establishes a TCP connection with the LDAP authorization server.
4. To obtain the right to search, the LDAP client uses the administrator DN and password to send an administrator bind request to the LDAP server.
5. The LDAP server processes the request. If the bind operation is successful, the LDAP server sends an acknowledgment to the LDAP client.

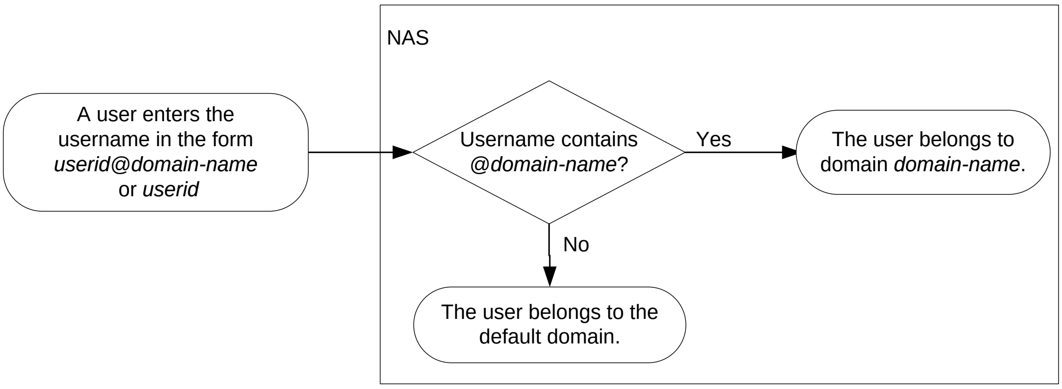
6. The LDAP client sends an authorization search request with the username of the Telnet user to the LDAP server. If the user uses the same LDAP server for authentication and authorization, the client sends the request with the saved user DN of the Telnet user to the LDAP server.
7. After receiving the request, the LDAP server searches for the user information by the base DN, search scope, filtering conditions, and LDAP attributes. If a match is found, the LDAP server sends a response to notify the LDAP client of the successful search.
8. After successful authorization, the LDAP client notifies the user of the successful login.

User management based on ISP domains and user access types

AAA manages users based on the users' ISP domains and access types.

On a NAS, each user belongs to one ISP domain. The NAS determines the ISP domain to which a user belongs based on the username entered by the user at login.

Figure 9 Determining the ISP domain for a user by username


NAS
A user enters the username in the form *userid@domain-name* or *userid*
Username contains *@domain-name*?
Yes
The user belongs to domain *domain-name*.
No
The user belongs to the default domain.

AAA manages users in the same ISP domain based on the users' access types. The device supports the following user access types:

- **LAN**—LAN users must pass MAC authentication to come online.
- **Login**—Login users include SSH, Telnet, FTP, and terminal users that log in to the device. Terminal users can access through a console port.
- ADVPN.
- **Portal**—Portal users must pass portal authentication to access the network.
- PPP.
- **IPoE**—IPoE users include Layer 2 and Layer 3 leased line users and Set Top Box (STB) users.
- **IKE**—IKE users must pass IKE extended authentication to access the network.
- **HTTP/HTTPS**—Users log in to the device through HTTP or HTTPS.
- SSL VPN.

Authentication, authorization, and accounting methods

AAA supports configuring different authentication, authorization, and accounting methods for different types of users in an ISP domain. The NAS determines the ISP domain and access type of a user. The NAS also uses the methods configured for the access type in the domain to control the user's access.

AAA also supports configuring a set of default methods for an ISP domain. These default methods are applied to users for whom no AAA methods are configured.

Authentication methods

The device supports the following authentication methods:

- **No authentication**—This method trusts all users and does not perform authentication. For security purposes, do not use this method.
- **Local authentication**—The NAS authenticates users by itself, based on the locally configured user information including the usernames, passwords, and attributes. Local authentication allows high speed and low cost, but the amount of information that can be stored is limited by the size of the storage space.
- **Remote authentication**—The NAS works with a remote server to authenticate users. The NAS communicates with the remote server through the RADIUS, LDAP, or HWTACACS protocol. The server manages user information in a centralized manner. Remote authentication provides high capacity, reliable, and centralized authentication services for multiple NASs. You can configure backup methods to be used when the remote server is not available.

Authorization methods

The device supports the following authorization methods:

- **No authorization**—The NAS performs no authorization exchange. The following default authorization information applies after users pass authentication:
  - Login users obtain the level-0 user role. For more information about the level-0 user role, see RBAC configuration in *Fundamentals Configuration Guide*.
  - The working directory for FTP, SFTP, and SCP login users is the root directory of the NAS. However, the users do not have permission to access the root directory.
  - Non-login users can access the network.
- **Local authorization**—The NAS performs authorization according to the user attributes locally configured for users.
- **Remote authorization**—The NAS works with a remote server to authorize users. RADIUS authorization is bound with RADIUS authentication. RADIUS authorization can work only after RADIUS authentication is successful, and the authorization information is included in the Access-Accept packet. HWTACACS or LDAP authorization is separate from authentication, and the authorization information is included in the authorization response after successful authentication. You can configure backup methods to be used when the remote server is not available.

Accounting methods

The device supports the following accounting methods:

- **No accounting**—The NAS does not perform accounting for the users.
- **Local accounting**—Local accounting is implemented on the NAS. It counts and controls the number of concurrent users that use the same local user account, but does not provide statistics for charging.
- **Remote accounting**—The NAS works with a RADIUS server or HWTACACS server for accounting. You can configure backup methods to be used when the remote server is not available.

AAA extended functions

The device provides the following login services to enhance device security:

- **Command authorization**—Enables the NAS to let the authorization server determine whether a command entered by a login user is permitted. Login users can execute only commands permitted by the authorization server. For more information about command authorization, see controlling user access to the device in *Fundamentals Configuration Guide*.
- **Command accounting**—When command authorization is disabled, command accounting enables the accounting server to record all valid commands executed on the device. When

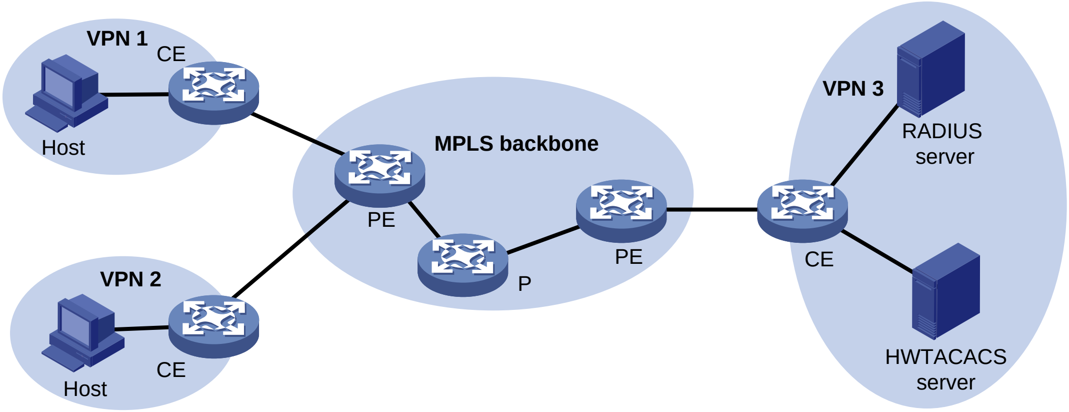
command authorization is enabled, command accounting enables the accounting server to record all authorized commands. For more information about command accounting, see controlling user access to the device in *Fundamentals Configuration Guide*.

- **User role authentication**—Authenticates each user that wants to obtain another user role without logging out or getting disconnected. For more information about user role authentication, see *Fundamentals Configuration Guide*.

AAA for VPNs

You can deploy AAA across VPNs to enable forwarding of authentication, authorization, and accounting packets across VPNs. For example, as shown in Figure 10, the PE at the left side of the MPLS backbone acts as a NAS. The NAS transparently delivers the AAA packets of private users in VPN 1 and VPN 2 to the AAA servers in VPN 3 for centralized authentication. Authentication packets of private users in different VPNs do not affect each other.

Figure 10 Network diagram



Protocols and standards

- RFC 2865, *Remote Authentication Dial In User Service (RADIUS)*
- RFC 2866, *RADIUS Accounting*
- RFC 2867, *RADIUS Accounting Modifications for Tunnel Protocol Support*
- RFC 2868, *RADIUS Attributes for Tunnel Protocol Support*
- RFC 2869, *RADIUS Extensions*
- RFC 3576, *Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)*
- RFC 4818, *RADIUS Delegated-IPv6-Prefix Attribute*
- RFC 5176, *Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)*
- RFC 1492, *An Access Control Protocol, Sometimes Called TACACS*
- RFC 1777, *Lightweight Directory Access Protocol*
- RFC 2251, *Lightweight Directory Access Protocol (v3)*

AAA tasks at a glance

To configure AAA, perform the following tasks:

1. Configuring AAA schemes

   If local authentication is used, configure local users and the related attributes. If remote authentication is used, configure the required RADIUS, LDAP, or HWTACACS schemes.

   - Configuring local users
   - Configuring RADIUS
   - Configuring HWTACACS
   - Configuring LDAP
2. Configuring an ISP domain
   a. Creating an ISP domain
   b. Configuring ISP domain attributes
3. Configuring AAA methods for an ISP domain

   Configure authentication, authorization, and accounting methods for an ISP domain as needed. These methods use existing AAA schemes.

   - Configuring authentication methods for an ISP domain
   - Configuring authorization methods for an ISP domain
   - Configuring accounting methods for an ISP domain
4. (Optional.) Configuring advanced AAA features
   - Configuring domain name delimiters
   - Setting the maximum number of concurrent login users
   - Configuring a NAS-ID
   - Enabling password change prompt logging
   - Configuring the device ID
   - Configuring the AAA test feature

Configuring local users

About local users

To implement local authentication, authorization, and accounting, create local users and configure user attributes on the device. The local users and attributes are stored in the local user database on the device. A local user is uniquely identified by the combination of a username and a user type.

Local users are classified into the following types:

- **Device management user**—User that logs in to the device for device management.
- **Network access user**—User that accesses network resources through the device.

  Network access users also include guests that access the network temporarily. Guests can use only LAN and portal services.

The following shows the configurable local user attributes:

- **Description**—Descriptive information of the user.
- **Service type**—Services that the user can use. Local authentication checks the service types of a local user. If none of the service types is available, the user cannot pass authentication.
- **User state**—Whether or not a local user can request network services. There are two user states: active and blocked. A user in active state can request network services, but a user in blocked state cannot.
- **Upper limit of concurrent logins using the same user name**—Maximum number of users that can concurrently access the device by using the same user name. When the number reaches the upper limit, no more local users can access the device by using the user name.

- **User group**—Each local user belongs to a local user group and has all attributes of the group. The attributes include the password control attributes and authorization attributes. For more information about local user group, see "Configuring user group attributes."
- **Identity group**—The user identification module controls the network access of a local user based on the identity group to which the user belongs. For more information about identity-based user identification, see "Configuring user identification."
- **Binding attributes**—Binding attributes control the scope of users, and are checked during local authentication of a user. If the attributes of a user do not match the binding attributes configured for the local user account, the user cannot pass authentication.
- **Authorization attributes**—Authorization attributes indicate the user's rights after it passes local authentication.

  Configure the authorization attributes based on the service type of local users. For example, you do not need to configure the FTP/SFTP/SCP working directory attribute for a PPP user.

  You can configure an authorization attribute in user group view or local user view. The setting of an authorization attribute in local user view takes precedence over the attribute setting in user group view.

The attribute configured in user group view takes effect on all local users in the user group.

The attribute configured in local user view takes effect only on the local user.

- **Password control attributes**—Password control attributes help control password security for local users. Password control attributes include password aging time, minimum password length, password composition checking, password complexity checking, and login attempt limit.

  You can configure a password control attribute in system view, user group view, or local user view. A password control attribute with a smaller effective range has a higher priority. For more information about password management and global password configuration, see "Configuring password control."
- **Validity period**—Time period in which a network access user is considered valid for authentication.

Local user configuration tasks at a glance

To configure local users, perform the following tasks:

1. Configuring local user attributes
   - Configuring attributes for device management users
   - Configuring attributes for network access users
   - Configuring local guest attributes
2. (Optional.) Configuring user group attributes
3. (Optional.) Configuring local users in batch
4. (Optional.) Managing network access users
5. (Optional.) Managing local guests

Configuring attributes for device management users

Restrictions and guidelines

The enabling status of global password control for device management users affects the way that the device stores the passwords of device management users.

- When password control is enabled globally for device management users, the device neither displays the passwords of the users nor retains the passwords in the running configuration.
- When you globally disable password control for device manangement users, the device automatically restores the passwords of the users to the running configuration.

To enable password control globally for device management users, use the **password-control enable** command. To display the running configuration, use the **display current-configuration** command.

You can configure authorization attributes and password control attributes in local user view or user group view. The setting in local user view takes precedence over the setting in user group view.

Procedure

1. Enter system view.

   system-view
2. Add a device management user and enter device management user view.

   local-user *user-name* [**class manage**]
3. Configure a password for the device management user.

   password [{ **hash** | **simple** } *string*]

   A non-password-protected user passes authentication if the user provides the correct username and passes attribute checks. To enhance security, configure a password for each device management user.
4. Assign services to the device management user.

   service-type { **ftp** | { **http** | **https** | **ssh** | **telnet** | **terminal** } * }

   By default, no services are authorized to a device management user.
5. (Optional.) Set the status of the device management user.

   state { **active** | **block** }

   By default, a device management user is in active state and can request network services.
6. (Optional.) Set the upper limit of concurrent logins using the device management username.

   access-limit *max-user-number*

   By default, the number of concurrent logins is not limited for a device management user.

   This command takes effect only when local accounting is configured for device management users. This command does not apply to FTP, SFTP, or SCP users that do not support accounting.
7. (Optional.) Configure authorization attributes for the device management user.

   authorization-attribute { **idle-cut** *minutes* | **user-role** *role-name* | **work-directory** *directory-name* } *

   The following default settings apply:
   - The working directory for FTP, SFTP, and SCP users is the root directory of the NAS. However, the users do not have permission to access the root directory.
   - The network-operator user role is assigned to local users that are created by a network-admin or level-15 user on the default context. The context-operator user role is assigned to local users that are created by a context-admin or level-15 user on a non-default context.
8. (Optional.) Configure password control attributes for the device management user. Choose the following tasks as needed:
   - Set the password aging time.

     password-control aging *aging-time*
   - Set the minimum password length.

     password-control length *length*
   - Configure the password composition policy.

     password-control composition type-number *type-number* [**type-length** *type-length*]

  - Configure the password complexity checking policy.

    password-control complexity { **same-character** | **user-name** } **check**
  - Configure the maximum login attempts and the action to take if there is a login failure.

    password-control login-attempt *login-times* [**exceed** { **lock** | **lock-time** *time* | **unlock** }]

   By default, a device management user uses password control attributes of the user group to which the user belongs.

9. (Optional.) Assign the device management user to a user group.

   group *group-name*

   By default, a device management user belongs to user group **system**.

Configuring attributes for network access users

Restrictions and guidelines

The enabling status of global password control for network access users affects the way that the device stores the passwords of device management users.

- When password control is enabled globally for network access users, the device neither displays the passwords of the users nor retains the passwords in the running configuration.
- When you globally disable password control for network access users, the device automatically restores the passwords of the users to the running configuration.

To enable password control globally for network access users, use the **password-control enable network-class** command. To display the running configuration, use the **display current-configuration** command.

You can configure authorization attributes and password control attributes in local user view or user group view. The setting in local user view takes precedence over the setting in user group view.

Configure the **location** binding attribute based on the service types of users.

- For portal users, specify the portal-enabled interfaces through which the users access the device. Specify the Layer 2 Ethernet interfaces if portal is enabled on VLAN interfaces and the **portal roaming enable** command is not used.

Procedure

1. Enter system view.

   system-view

2. Add a network access user and enter network access user view.

   local-user *user-name* **class network**

3. (Optional.) Configure a password for the network access user.

   password { **cipher** | **simple** } *string*

4. (Optional.) Configure a description for the network access user.

   description *text*

   By default, no description is configured for a local user.

5. Assign services to the network access user.

   service-type { **advpn** | **ike** | **ipoe** | **lan-access** | **portal** | **ppp** | **sslvpn** }

   By default, no services are authorized to a network access user.

6. (Optional.) Set the status of the network access user.

   state { **active** | **block** }

   By default, a network access user is in active state and can request network services.

7. (Optional.) Set the upper limit of concurrent logins using the network access username.

   access-limit *max-user-number*

   By default, the number of concurrent logins is not limited for a network access user.

8. (Optional.) Configure binding attributes for the network access user.

   bind-attribute { **call-number** *call-number* [**:** *subcall-number*] | **location interface** *interface-type interface-number* | **mac** *mac-address* | **vlan** *vlan-id* } *

   By default, no binding attributes are configured for a network access user.

9. (Optional.) Configure authorization attributes for the network access user.

   authorization-attribute { **acl** *acl-number* | **callback-number** *callback-number* | **idle-cut** *minutes* | **ip** *ipv4-address* | **ip-pool** *ipv4-pool-name* | **ipv6** *ipv6-address* | **ipv6-pool** *ipv6-pool-name* | **ipv6-prefix** *ipv6-prefix prefix-length* | { **primary-dns** | **secondary-dns** } { **ip** *ipv4-address* | **ipv6** *ipv6-address* } | **session-timeout** *minutes* | **sslvpn-policy-group** *group-name* | **url** *url-string* | **vlan** *vlan-id* | **vpn-instance** *vpn-instance-name* } *

   By default, a network access user does not have authorization attributes.

10. (Optional.) Configure password control attributes for the network access user. Choose the following tasks as needed:

    - Set the minimum password length.

      password-control length *length*

    - Configure the password composition policy.

      password-control composition type-number *type-number* [**type-length** *type-length*]

    - Configure the password complexity checking policy.

      password-control complexity { **same-character** | **user-name** } **check**

    By default, a network access user uses password control attributes of the user group to which the user belongs.

11. (Optional.) Assign the network access user to a user group.

    group *group-name*

    By default, a network access user belongs to user group **system**.

12. (Optional.) Assign the network access user to an identity group.

    identity-group *group-name*

    By default, a network access user does not belong to any identity groups.

    You can execute this command multiple times to add a network access user to multiple identity groups. After you add a network access user to an identity group, the system automatically synchronizes the configuration to the identity group. Then, the user is available in user group view of the identity group.

13. (Optional.) specify the validity period for the local user.

    validity-datetime { **from** *start-date start-time* **to** *expiration-date expiration-time* | **from** *start-date start-time* | **to** *expiration-date expiration-time* }

    By default, the validity period for a network access user does not expire.

Configuring local guest attributes

About this task

Create local guests and configure guest attributes to control temporary network access behavior. Guests can access the network after passing local authentication. You can configure the recipient addresses and email attribute information to the local guests and guest sponsors.

Procedure

1. Enter system view.

   system-view

2. Create a local guest and enter local guest view.

   local-user *user-name* **class network guest**

3. (Optional.) Configure a password for the local guest.

   password { **cipher** | **simple** } *string*

4. Configure basic information for the local guest. Choose the following tasks as needed:

   - Configure a description for the local guest.

     description *text*

     By default, no description is configured for a local guest.

   - Specify the name of the local guest.

     full-name *name-string*

     By default, no name is specified for a local guest.

   - Specify the company of the local guest.

     company *company-name*

     By default, no company is specified for a local guest.

   - Specify the phone number of the local guest.

     phone *phone-number*

     By default, no phone number is specified for a local guest.

   - Specify the email address of the local guest.

     email *email-string*

     By default, no email address is specified for a local guest.

   - Specify the sponsor name for the local guest.

     sponsor-full-name *name-string*

     By default, no sponsor name is specified for a local guest.

   - Specify the sponsor department for the local guest.

     sponsor-department *department-string*

     By default, no sponsor department is specified for a local guest.

   - Specify the sponsor email address for the local guest.

     sponsor-email *email-string*

     By default, no sponsor email address is specified for a local guest.

5. (Optional.) Configure the validity period for the local guest.

   validity-datetime from *start-date start-time* **to** *expiration-date expiration-time*

   By default, a local guest does not expire.

6. (Optional.) Assign the local guest to a user group.

   group *group-name*

By default, a local guest belongs to the system-defined user group **system**.

7. (Optional.) Configure the local guest status.

   state { **active** | **block** }

   By default, a local guest is in active state and is allowed to request network services.

Configuring user group attributes

About this task

User groups simplify local user configuration and management. A user group contains a group of local users and has a set of local user attributes. You can configure local user attributes for a user group to implement centralized user attributes management for the local users in the group. Local user attributes that are manageable include authorization attributes.

Procedure

1. Enter system view.

   system-view

2. Create a user group and enter user group view.

   user-group *group-name*

   By default, a system-defined user group exists. The group name is **system**.

3. Configure authorization attributes for the user group.

   authorization-attribute { **acl** *acl-number* | **callback-number** *callback-number* | **idle-cut** *minutes* | **ip-pool** *ipv4-pool-name* | **ipv6-pool** *ipv6-pool-name* | **ipv6-prefix** *ipv6-prefix prefix-length* | { **primary-dns** | **secondary-dns** } { **ip** *ipv4-address* | **ipv6** *ipv6-address* } | **session-timeout** *minutes* | **sslvpn-policy-group** *group-name* | **url** *url-string* | **vlan** *vlan-id* | **vpn-instance** *vpn-instance-name* | **work-directory** *directory-name* } *

   By default, no authorization attributes are configured for a user group.

4. (Optional.) Configure password control attributes for the user group. Choose the following tasks as needed:

   - Set the password aging time.

     password-control aging *aging-time*

   - Set the minimum password length.

     password-control length *length*

   - Configure the password composition policy.

     password-control composition type-number *type-number* [**type-length** *type-length*]

   - Configure the password complexity checking policy.

     password-control complexity { **same-character** | **user-name** } **check**

   - Configure the maximum login attempts and the action to take for login failures.

     password-control login-attempt *login-times* [**exceed** { **lock** | **lock-time** *time* | **unlock** }]

   By default, a user group uses the global password control settings. For more information, see "Configuring password control."

5. (Optional.) Assign identity members to the user group.

   identity-member { **group** *group-name* | **user** *user-name* }

   By default, no identity members exist in a user group.

   You cannot add a user group to a lower-level group that is an identity member of the group.

Configuring local users in batch

About this task

You can import or export local network access users in batch.

Importing local users in batch

1. Enter system view.

   system-view

2. Import user account information from a .csv file in the specified path to create network access users based on the imported information.

   local-user-import class network url *url-string* [**auto-create-group** | **override** | **start-line** *line-number*] *

   Make sure the .csv file contains at least the user name for each account. If other parameters are not specified for accounts in the file, the system automatically assigns them default values.

Exporting local users in batch

1. Enter system view.

   system-view

2. Export network access user account information to a .csv file in the specified path.

   local-user-export class network url *url-string* [**from** { **group** *group-name* | **user** *user-name* }]

   Before the import, you can edit the file as needed. For information about the editing restrictions, see *Security Command Reference*.

Managing network access users

About this task

You can configure the device to generate a random password for a network access user on the Web interface. The device supports sending the random password and other user account information to the user by email. The email and email server parameters can be configured from the Web interface or at the CLI.

Perform the tasks in this section to configure the email and email server parameters at the CLI.

Restrictions and guidelines

Make sure the device and the SMTP email server have routes to reach each other.

Make sure the user emails configured on the Web interface are valid and reachable.

The configurations on the Web interface and CLI for the same attribute will overwrite each other.

Procedure

1. Enter system view.

   system-view

2. Configure parameters for the email notifications sent to network access users.

   - Configure the email sender address.

     access-user email sender *email-address*

     By default, no email sender address is configured.

   - Configure the body and subject of the email notifications sent to network access users.

     access-user email format { **body** *body-string* | **subject** *sub-string* }

By default, the email subject is **Password reset notification**, and the email body is as follows:

```
A random password has been generated for your account.
Username: xxx
Password: yyy
Validity: YYYY/MM/DD hh:mm:ss to YYYY/MM/DD hh:mm:ss
```

3. Configure SMTP server parameters for the email notifications sent to network access users.
   - Specify an SMTP server to send email notifications to network access users.

     access-user email smtp-server *url-string*

     By default, no SMTP server is specified to send email notifications to network access users.
   - Configure the username and password of the SMTP server.

     access-user email authentication username *user-name* **password** { **cipher** | **simple** } *string*

     By default, no username or password is configured for logging in to an SMTP server.

     If the SMTP server requires a username and password for login, you must configure the username and password on the device for the server.

Managing local guests

About this task

The local guest management features are for registration, approval, maintenance, and access control of local guests.

The registration and approval processes are as follows:

1. The device pushes the portal user registration page to a user that wants to access the network as a local guest.
2. The user submits account information for registration, including the user name, password, and email address.
3. The device forwards the registration request to the guest manager in an email notification.
4. The guest manager adds supplementary information as needed and approves the registration information.

   The guest manager must process the registration request before the waiting-approval timeout timer expires. The device automatically deletes expired registration request information.
5. The device creates a local guest account and sends an email notification to the user and guest sponsor. The email contains local guest account, password, validity period, and other account information.

   The user can access the network as a local guest.

The device provides the following local guest management features:

- **Registration and approval**—The device creates local guests after the guest registration information is approved by a guest manager.
- **Email notification**—The device notifies the local guests, guest sponsors, or guest managers by email of the guest account information or guest registration requests.
- **Local guest creation in batch**—Create a batch of local guests.
- **Local guest import**—Import guest account information from a .csv file to create local guests on the device based on the imported information.
- **Local guest export**—Export local guest account information to a .csv file. You can import the account information to other devices as needed.
- **Guest auto-delete**—The device checks the validity status of each local guest and automatically deletes expired local guests.

Procedure

1. Enter system view

   system-view

2. Configure the email notification feature for local guests.

   a. Configure the subject and body of email notifications.

      local-guest email format to { **guest** | **manager** | **sponsor** } { **body** *body-string* | **subject** *sub-string* }

      By default, no subject or body is configured.

   b. Configure the email sender address in the email notifications sent by the device for local guests.

      local-guest email sender *email-address*

      By default, no email sender address is configured for the email notifications sent by the device.

   c. Specify an SMTP server for sending email notifications of local guests.

      local-guest email smtp-server *url-string*

      By default, no SMTP server is specified.

3. Configure the guest manager's email address.

   local-guest manager-email *email-address*

   By default, the guest manager's email address is not configured.

4. (Optional.) Set the waiting-approval timeout timer for guest registration requests.

   local-guest timer waiting-approval *time-value*

   By default, the waiting-approval timeout timer for guest registration requests is 24 hours.

5. (Optional.) Import guest account information from a .csv file in the specified path to create local guests based on the imported information.

   local-user-import class network guest url *url-string* **validity-datetime** *start-date start-time* **to** *expiration-date expiration-time* [**auto-create-group** | **override** | **start-line** *line-number*] *

6. (Optional.) Create local guests in batch.

   local-guest generate username-prefix *name-prefix* [**password-prefix** *password-prefix*] **suffix** *suffix-number* [**group** *group-name*] **count** *user-count* **validity-datetime** *start-date start-time* **to** *expiration-date expiration-time*

   Batch generated local guests share the same name prefix. You can also configure a password prefix to be shared by the guests.

7. (Optional.) Export local guest account information to a .csv file in the specified path.

   local-user-export class network guest url *url-string*

8. (Optional.) Enable the guest auto-delete feature.

   local-guest auto-delete enable

   By default, the guest auto-delete feature is disabled.

9. (Optional.) Send email notifications to the local guest or the guest sponsor.

   a. Return to user view.

      quit

   b. Send email notifications to the local guest or the guest sponsor. The email contents include the user name, password, and validity period of the guest account.

      local-guest send-email user-name *user-name* **to** { **guest** | **sponsor** }

Display and maintenance commands for local users and local user groups

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display pending registration requests for local guests.	**display local-guest waiting-approval** [**user-name** *user-name*]
Display the local user configuration and online user statistics.	**display local-user** [**class** { **manage** | **network** [**guest**] } | **idle-cut** { **disable** | **enable** } | **service-type** { **advpn** | **ftp** | **http** | **https** | **ike** | **ipoe** | **lan-access** | **portal** | **ppp** | **ssh** | **sslvpn** | **telnet** | **terminal** } | **state** { **active** | **block** } | **user-name** *user-name* **class** { **manage** | **network** [**guest**] } | **vlan** *vlan-id*]
Display user group configuration.	**display user-group** { **all** | **name** *group-name* } [**identity-member** { **all** | **group** | **user** }]
Clear pending registration requests for local guests.	**reset local-guest waiting-approval** [**user-name** *user-name*]

Configuring RADIUS

RADIUS tasks at a glance

To configure RADIUS, perform the following tasks:

1. Configuring a test profile for RADIUS server status detection

   To detect the status of a RADIUS server, you must configure a test profile and configure the RADIUS server to use the test profile in a RADIUS scheme.
2. Creating a RADIUS scheme
3. Specifying RADIUS authentication servers
4. Specifying the RADIUS accounting servers
5. Specifying the shared keys for secure RADIUS communication

   Perform this task if no shared keys are specified when configuring RADIUS authentication or accounting servers.
6. Specifying the MPLS L3VPN instance for a RADIUS scheme

   Perform this task if no MPLS L3VPN instances are specified when configuring RADIUS authentication or accounting servers.
7. (Optional.) Setting the status of RADIUS servers
8. (Optional.) Setting RADIUS timers
9. (Optional.) Configuring parameters for RADIUS packets
   - Specifying the source IP address for outgoing RADIUS packets
   - Setting the username format and traffic statistics units
   - Setting the maximum number of RADIUS request transmission attempts
   - Setting the maximum number of real-time accounting attempts
   - Setting the DSCP priority for RADIUS packets

10. (Optional.) Configuring parameters for RADIUS attributes
    - Configuring the Login-Service attribute check method for SSH, FTP, and terminal users
    - Enabling online user password change by using RADIUS attribute 17
    - Configuring parsing rules for the RDIUS Reply-Message attribute
    - Interpreting the RADIUS class attribute as CAR parameters
    - Configuring the MAC address format for the RADIUS Called-Station-Id attribute
    - Configuring the MAC address format for the RADIUS Calling-Station-Id attribute
    - Specifying a server version for interoperating with servers with a vendor ID of 2011
    - Setting the data measurement unit for the Remanent_Volume attribute
    - Interpreting the Microsegment-Id attribute to an authorization VLAN
    - Specifying the format for attribute Acct-Session-Id
    - Configuring the RADIUS attribute translation feature
11. (Optional.) Configuring extended RADIUS features
    - Configuring the RADIUS accounting-on feature
    - Configuring the RADIUS session-control feature
    - Configuring the RADIUS DAS feature
    - Enabling SNMP notifications for RADIUS

Configuring a test profile for RADIUS server status detection

About this task

To detect the reachability of a RADIUS authentication server, specify a test profile for the RADIUS server when you specify the server in a RADIUS scheme. With the test profile, the device refreshes the RADIUS server status at each detection interval according to the detection result. If the server is unreachable, the device sets the status of the server to blocked. If the server is reachable, the device sets the status of the server to active.

After you specify an existing test profile, the device starts detecting the status of a RADIUS server by simulating an authentication request with the username specified in the test profile. The authentication request is sent to the RADIUS server within each detection interval. The device determines that the RADIUS server is reachable if the device receives a response from the server within the interval.

Restrictions and guidelines

You can configure multiple test profiles in the system.

The device stops detecting the status of a RADIUS server when one of the following operations is performed:

- The RADIUS server is removed from the RADIUS scheme.
- The test profile configuration for the RADIUS server is removed in RADIUS scheme view.
- The test profile specified for the RADIUS server is deleted.
- The RADIUS server is manually set to the blocked state.
- The RADIUS scheme that contains the RADIUS server is deleted.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure a test profile for detecting the status of RADIUS authentication servers.

radius-server test-profile *profile-name* **username** *name* [**interval** *interval*]

Creating a RADIUS scheme

Restrictions and guidelines

You can configure a maximum of 16 RADIUS schemes. A RADIUS scheme can be used by multiple ISP domains.

Procedure

1. Enter system view.

   system-view

2. Create a RADIUS scheme and enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

Specifying RADIUS authentication servers

About this task

A RADIUS authentication server completes authentication and authorization together, because authorization information is piggybacked in authentication responses sent to RADIUS clients.

You can specify one primary authentication server and a maximum of 16 secondary authentication servers for a RADIUS scheme. Secondary servers provide AAA services when the primary server becomes unreachable. The device searches for an active server in the order the secondary servers are configured.

Restrictions and guidelines

If redundancy is not required, specify only the primary server.

A RADIUS authentication server can function as the primary authentication server for one scheme and a secondary authentication server for another scheme at the same time.

Two authentication servers in a scheme, primary or secondary, cannot have the same combination of IP address, VPN instance, and port number.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Specify the primary RADIUS authentication server.

   primary authentication { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **test-profile** *profile-name* | **vpn-instance** *vpn-instance-name*] *

   By default, no primary RADIUS authentication server is specified.

4. (Optional.) Specify a secondary RADIUS authentication server.

   secondary authentication { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **test-profile** *profile-name* | **vpn-instance** *vpn-instance-name*] *

   By default, no secondary RADIUS authentication servers are specified.

Specifying the RADIUS accounting servers

About this task

You can specify one primary accounting server and a maximum of 16 secondary accounting servers for a RADIUS scheme. Secondary servers provide AAA services when the primary server becomes unavailable. The device searches for an active server in the order the secondary servers are configured.

Restrictions and guidelines

If redundancy is not required, specify only the primary server.

A RADIUS accounting server can function as the primary accounting server for one scheme and a secondary accounting server for another scheme at the same time.

Two accounting servers in a scheme, primary or secondary, cannot have the same combination of IP address, VPN instance, and port number.

RADIUS does not support accounting for FTP, SFTP, and SCP users.

Procedure

1. Enter system view.
   system-view
2. Enter RADIUS scheme view.
   radius scheme *radius-scheme-name*
3. Specify the primary RADIUS accounting server.
   primary accounting { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **vpn-instance** *vpn-instance-name*] *
   By default, no primary RADIUS accounting server is specified.
4. (Optional.) Specify a secondary RADIUS accounting server.
   secondary accounting { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **vpn-instance** *vpn-instance-name*] *
   By default, no secondary RADIUS accounting servers are specified.

Specifying the shared keys for secure RADIUS communication

About this task

The RADIUS client and server use the MD5 algorithm and shared keys to generate the Authenticator value for packet authentication and user password encryption. The client and server must use the same key for each type of communication.

A key configured in this task is for all servers of the same type (accounting or authentication) in the scheme. The key has a lower priority than a key configured individually for a RADIUS server.

Restrictions and guidelines

The shared key configured on the device must be the same as the shared key configured on the RADIUS server.

Procedure

1. Enter system view.
   system-view
2. Enter RADIUS scheme view.

radius scheme *radius-scheme-name*

3. Specify a shared key for secure RADIUS communication.

   key { **accounting** | **authentication** } { **cipher** | **simple** } *string*

   By default, no shared key is specified for secure RADIUS communication.

Specifying the MPLS L3VPN instance for a RADIUS scheme

About this task

The VPN instance specified for a RADIUS scheme applies to all authentication and accounting servers in that scheme. If a VPN instance is also configured for an individual RADIUS server, the VPN instance specified for the RADIUS scheme does not take effect on that server.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Specify a VPN instance for the RADIUS scheme.

   vpn-instance *vpn-instance-name*

   By default, a RADIUS scheme belongs to the public network.

Setting the status of RADIUS servers

About this task

To control the RADIUS servers with which the device communicates when the current servers are no longer available, set the status of RADIUS servers to blocked or active. You can specify one primary RADIUS server and multiple secondary RADIUS servers. The secondary servers function as the backup of the primary server. The device chooses servers based on the following rules:

- When the primary server is in active state, the device first tries to communicate with the primary server. If the primary server is unreachable, the device searches for an active secondary server in the order the servers are configured.
- When one or more servers are in active state, the device tries to communicate with these active servers only, even if the servers are unavailable.
- When all servers are in blocked state, the device tries to communicate with a server as follows:
  - If the primary server is placed in blocked state automatically, the device only tries to communicate with the primary server.
  - If the primary server is placed in blocked state manually, the device tries to communicate with secondary servers automatically placed in blocked state in the sequence they are configured.
- If a server is unreachable, the device performs the following operations:
  - Changes the server status to blocked.
  - Starts a quiet timer for the server.
  - Tries to communicate with the next secondary server in active state that has the highest priority.
- When the quiet timer of a server expires or you manually set the server to the active state, the status of the server changes back to active. The device does not check the server again during the authentication or accounting process.

- The search process continues until the device finds an available secondary server or has checked all secondary servers in active state. If no server is reachable, the device considers the authentication or accounting attempt a failure.
- When you remove a server in use, communication with the server times out. The device looks for a server in active state by first checking the primary server, and then checking secondary servers in the order they are configured.
- When a RADIUS server's status changes automatically, the device changes this server's status accordingly in all RADIUS schemes in which this server is specified.
- When a RADIUS server is manually set to blocked, server detection is disabled for the server, regardless of whether a test profile has been specified for the server. When the RADIUS server is set to active state, server detection is enabled for the server on which an existing test profile is specified.

By default, the device sets the status of all RADIUS servers to active. However, in some situations, you must change the status of a server. For example, if a server fails, you can change the status of the server to blocked to avoid communication attempts to the server.

Restrictions and guidelines

The configured server status cannot be saved to any configuration file, and can only be viewed by using the **display radius scheme** command.

After the device restarts, all servers are restored to the active state.

Procedure

1. Enter system view.
   system-view
2. Enter RADIUS scheme view.
   radius scheme *radius-scheme-name*
3. Set the RADIUS server status. Choose the following tasks as needed:
   - Set the status of the primary RADIUS authentication server.
     state primary authentication { **active** | **block** }
   - Set the status of the primary RADIUS accounting server.
     state primary accounting { **active** | **block** }
   - Set the status of a secondary RADIUS authentication server.
     state secondary authentication [{ *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **vpn-instance** *vpn-instance-name*] *] { **active** | **block** }
   - Set the status of a secondary RADIUS accounting server.
     state secondary accounting [{ *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **vpn-instance** *vpn-instance-name*] *] { **active** | **block** }

   By default, a RADIUS server is in active state.

Setting RADIUS timers

About this task

The device uses the following types of timers to control communication with a RADIUS server:

- **Server response timeout timer** (**response-timeout**)—Defines the RADIUS request retransmission interval. The timer starts immediately after a RADIUS request is sent. If the device does not receive a response from the RADIUS server before the timer expires, it resends the request.

- **Server quiet timer** (**quiet**)—Defines the duration to keep an unreachable server in blocked state. If one server is not reachable, the device changes the server status to blocked, starts this timer for the server, and tries to communicate with another server in active state. After the server quiet timer expires, the device changes the status of the server back to active.
- **Real-time accounting timer** (**realtime-accounting**)—Defines the interval at which the device sends real-time accounting packets to the RADIUS accounting server for online users.

Restrictions and guidelines

Consider the number of secondary servers when you configure the maximum number of RADIUS packet transmission attempts and the RADIUS server response timeout timer. If the RADIUS scheme includes many secondary servers, the retransmission process might be too long and the client connection in the access module, such as Telnet, can time out.

When the client connections have a short timeout period, a large number of secondary servers can cause the initial authentication or accounting attempt to fail. In this case, reconnect the client rather than adjusting the RADIUS packet transmission attempts and server response timeout timer. Typically, the next attempt will succeed, because the device has blocked the unreachable servers to shorten the time to find a reachable server.

Make sure the server quiet timer is set correctly. A timer that is too short might result in frequent authentication or accounting failures. This is because the device will continue to attempt to communicate with an unreachable server that is in active state. A timer that is too long might temporarily block a reachable server that has recovered from a failure. This is because the server will remain in blocked state until the timer expires.

A short real-time accounting interval helps improve accounting precision but requires many system resources. When there are 1000 or more users, set the interval to 15 minutes or longer.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enter RADIUS scheme view.
   ```
   radius scheme radius-scheme-name
   ```
3. Set RADIUS timers. Choose the following tasks as needed:
   - Set the RADIUS server response timeout timer.
     ```
     timer response-timeout seconds
     ```
     The default setting is 3 seconds.
   - Set the quiet timer for the servers.
     ```
     timer quiet minutes
     ```
     The default setting is 5 minutes.
   - Set the real-time accounting timer.
     ```
     timer realtime-accounting interval [ second ]
     ```
     The default setting is 12 minutes.

Specifying the source IP address for outgoing RADIUS packets

About this task

The source IP address of RADIUS packets that a NAS sends must match the IP address of the NAS configured on the RADIUS server. A RADIUS server identifies a NAS by its IP address. Upon receiving a RADIUS packet, the RADIUS server checks the source IP address of the packet.

- If it is the IP address of a managed NAS, the server processes the packet.

- If it is not the IP address of a managed NAS, the server drops the packet.

Before sending a RADIUS packet, the NAS selects a source IP address in the following order:

1. The source IP address specified for the RADIUS scheme.
2. The source IP address specified in system view for the VPN or public network, depending on where the RADIUS server resides.
3. The IP address of the outbound interface specified by the route.

Restrictions and guidelines for source IP address configuration

You can specify a source IP address for outgoing RADIUS packets in RADIUS scheme view or in system view.

- The IP address specified in RADIUS scheme view applies only to one RADIUS scheme.
- The IP address specified in system view applies to all RADIUS schemes.

The source IP address of RADIUS packets that a NAS sends must match the IP address of the NAS that is configured on the RADIUS server.

As a best practice, specify a loopback interface address as the source IP address for outgoing RADIUS packets to avoid RADIUS packet loss caused by physical port errors.

The source address of outgoing RADIUS packets is typically the IP address of an egress interface on the NAS to communicate with the RADIUS server.

Specifying a source IP address for all RADIUS schemes

1. Enter system view.

   system-view

2. Specify a source IP address for outgoing RADIUS packets.

   radius nas-ip { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]

   By default, the source IP address of an outgoing RADIUS packet is the primary IPv4 address or the IPv6 address of the outbound interface.

Specifying a source IP address for a RADIUS scheme

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Specify a source IP address for outgoing RADIUS packets.

   nas-ip { *ipv4-address* | **ipv6** *ipv6-address* }

   By default, the source IP address of an outgoing RADIUS packet is that specified by using the **radius nas-ip** command in system view. If the **radius nas-ip** command is not used, the source IP address is the primary IP address of the outbound interface.

Setting the username format and traffic statistics units

About this task

A username is in the *userid@isp-name* format, where the *isp-name* part represents the user's ISP domain name. By default, the ISP domain name is included in a username. However, older RADIUS servers might not recognize usernames that contain the ISP domain names. In this case, you can configure the device to remove the domain name of each username to be sent.

The device reports online user traffic statistics in accounting packets. The traffic measurement units are configurable.

Restrictions and guidelines

If two or more ISP domains use the same RADIUS scheme, configure the RADIUS scheme to keep the ISP domain name in usernames for domain identification.

For accounting accuracy, make sure the traffic statistics units configured on the device and on the RADIUS accounting servers are the same.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Set the format for usernames sent to the RADIUS servers.

   user-name-format { **keep-original** | **with-domain** | **without-domain** }

   By default, the ISP domain name is included in a username.

4. Set the data flow and packet measurement units for traffic statistics.

   data-flow-format { **data** { **byte** | **giga-byte** | **kilo-byte** | **mega-byte** } | **packet** { **giga-packet** | **kilo-packet** | **mega-packet** | **one-packet** } }*

   By default, traffic is counted in bytes and packets.

Setting the maximum number of RADIUS request transmission attempts

About this task

RADIUS uses UDP packets to transfer data. Because UDP communication is not reliable, RADIUS uses a retransmission mechanism to improve reliability. A RADIUS request is retransmitted if the NAS does not receive a server response for the request within the response timeout timer. For more information about the RADIUS server response timeout timer, see "Setting the status of RADIUS servers."

You can set the maximum number for the NAS to retransmit a RADIUS request to the same server. When the maximum number is reached, the NAS tries to communicate with other RADIUS servers in active state. If no other servers are in active state at the time, the NAS considers the authentication or accounting attempt a failure.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Set the maximum number of RADIUS request transmission attempts.

   retry *retries*

   By default, the maximum number is 3 for RADIUS request transmission attempts.

Setting the maximum number of real-time accounting attempts

About this task

If you set the maximum number of real-time accounting attempts, the device will disconnect users from whom no accounting responses are received within the permitted attempts.

Procedure

1. Enter system view.

   `system-view`

2. Enter RADIUS scheme view.

   `radius scheme` *`radius-scheme-name`*

3. Set the maximum number of real-time accounting attempts.

   `retry realtime-accounting` *`retries`*

   By default, the maximum number is 5 for real-time accounting attempts.

Setting the DSCP priority for RADIUS packets

About this task

The DSCP priority in the ToS field determines the transmission priority of RADIUS packets. A larger value represents a higher priority.

Procedure

1. Enter system view.

   `system-view`

2. Set the DSCP priority for RADIUS packets.

   `radius` [**`ipv6`**] **`dscp`** *`dscp-value`*

   By default, the DSCP priority is 0 for RADIUS packets.

Configuring the Login-Service attribute check method for SSH, FTP, and terminal users

About this task

The device supports the following check methods for the Login-Service attribute (RADIUS attribute 15) of SSH, FTP, and terminal users:

- **Strict**—Matches Login-Service attribute values 50, 51, and 52 for SSH, FTP, and terminal services, respectively.
- **Loose**—Matches the standard Login-Service attribute value 0 for SSH, FTP, and terminal services.

An Access-Accept packet received for a user must contain the matching attribute value. Otherwise, the user cannot log in to the device.

Restrictions and guidelines

Use the loose check method only when the server does not issue Login-Service attribute values 50, 51, and 52 for SSH, FTP, and terminal users.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Configure the Login-Service attribute check method for SSH, FTP, and terminal users.

   attribute 15 check-mode { **loose** | **strict** }

   The default check method is strict.

Enabling online user password change by using RADIUS attribute 17

About this task

This feature enables the device to cooperate with the RADIUS server to allow users to change their passwords online.

When this feature is enabled, the process of online password change is as follows for a user when the user passes authentication:

1. If the RADIUS authentication server sends an Access-Challenge packet that includes the Reply-Message attribute when a user passes authentication, the device prompts the user to change its password.
2. After receiving the password change request from the user, the device sends a RADIUS authentication request to the RADIUS authentication server.

   In the authentication request, the device uses attribute 2 to carry the new user password.
3. When the device receives a response from the RADIUS authentication server, the online user's password is changed successfully.

When this feature is enabled, the process of online password change is as follows for an online user:

1. After receiving the password change request from the user, the device sends a RADIUS authentication request to a reachable RADIUS server. In the authentication request, the device uses attribute 2 and attribute 17 to carry the new user password and old user password, respectively.

   The RADIUS server selection process for online password change is the same as the process used to select a RADIUS authentication server. Online password change might fail because the device selects a RADIUS server different from the RADIUS server that authenticated the user.
2. When the device receives a response from the selected RADIUS server, the online user's password is changed successfully.

Restrictions and guidelines

This feature is applicable only to SSL VPN users.

Do not enable this feature if the RADIUS server does not support online user password change.

In a RADIUS scheme with this feature enabled, do not configure parsing rules for the Reply-Message attribute by using the **attribute 18 match** command. A violation will cause this feature fail to take effect.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Enable online user password change by using RADIUS attribute 17.

   attribute 17 old-password

   By default, online user password change is disabled.

Configuring parsing rules for the RDIUS Reply-Message attribute

About this task

The RADIUS Reply-Message attribute (attribute 18) is intended for the RADIUS server to return a message to users. In the Access-Challenge packets, this attribute indicates the action that the RADIUS server expects users to take. For the access device to correctly parse this attribute, you can configure parsing rules for this attribute. For example, the device needs to parse the Reply-Message attribute containing the **new pin** string as to prompt users to change the passwords online.

Each parsing rule contains a match criterion and an action. The device uses the fuzzy match method to match the Reply-Message attribute value against the match criterion. If the attribute value partially matches the match criterion, the device prompts the users to take the action specified in the parsing rule. Supported actions include the following:

- **new-password**—Enters the new password.
- **next-token**—Enters the next authentication factor for double-factor authentication.

Restirctions and guidelines

This feature is applicable only to SSL VPN users.

Before you configure parsing rules, make sure you fully understand the implications of the Reply-Message attribute defined by the RADIUS server.

For a RADIUS scheme, you can configure a maximum of 18 parsing rules for the Reply-Message attribute. Make sure the match criterion in each parsing rule is not contained by the match criterion of another parsing rule.

When parsing rules for the Reply-Message attribute are configured, the online user password change feature configured by using the **attribute 17 old-password** command does not take effect. As a best practice, do not configure both parsing rules and online user password change in the same RADIUS scheme.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Configure a parsing rule for the RADIUS Reply-Message attribute.

   attribute 18 match *string* **action** { **new-password** | **next-token** }

   By default, no parsing rules for the RADIUS Reply-Message attribute are configured. The device parses the Reply-Message attribute as to prompt users to enter the next authentication factor for double-factor authentication.

Interpreting the RADIUS class attribute as CAR parameters

About this task

A RADIUS server may deliver CAR parameters for user-based traffic monitoring and control by using the RADIUS class attribute (attribute 25) in RADIUS packets. You can configure the device to interpret the class attribute to CAR parameters.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Interpret the RADIUS class attribute as CAR parameters.

   attribute 25 car

   By default, the RADIUS class attribute is not interpreted as CAR parameters.

Configuring the MAC address format for the RADIUS Called-Station-Id attribute

Restrictions and guidelines

RADIUS servers of different types might have different requirements for the MAC address format in the RADIUS Called-Station-Id attribute. Configure the MAC address format for this attribute to meet the requirements of the RADIUS servers.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Configure the MAC address format for the RADIUS Called-Station-Id attribute.

   attribute 30 mac-format section { **one** | { **six** | **three** } **separator** *separator-character* } { **lowercase** | **uppercase** }

   By default, the MAC address in the RADIUS Called-Station-Id attribute is in the format of HH-HH-HH-HH-HH-HH. The MAC address is separated by hyphen (-) into six sections with letters in upper case.

Configuring the MAC address format for the RADIUS Calling-Station-Id attribute

Restrictions and guidelines

RADIUS servers of different types might have different requirements for the MAC address format in the RADIUS Calling-Station-Id attribute (attribute 31). Configure the MAC address format for this attribute to meet the requirements of the RADIUS servers.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Configure the MAC address format for the RADIUS Calling-Station-Id attribute.

   attribute 31 mac-format section { **one** | { **six** | **three** } **separator** *separator-character* } { **lowercase** | **uppercase** }

By default, the MAC address in the RADIUS Calling-Station-Id attribute is in the format of HH-HH-HH-HH-HH-HH. The MAC address is separated by hyphen (-) into six sections with letters in upper case.

Specifying a server version for interoperating with servers with a vendor ID of 2011

About this task

For the device to correctly interpret RADIUS attributes from the servers with a vendor ID of 2011, specify a server version that is the same as the version of the RADIUS servers.

Procedure

1. Enter system view.

   `system-view`

2. Enter RADIUS scheme view.

   `radius scheme` *`radius-scheme-name`*

3. Specify a server version for interoperating with servers with a vendor ID of 2011.

   `attribute vendor-id 2011 version` { **`1.0`** | **`1.1`** }

   By default, version 1.0 is used.

Setting the data measurement unit for the Remanent_Volume attribute

About this task

The RADIUS server uses the Remanent_Volume attribute in authentication or real-time accounting responses to notify the device of the current amount of data available for online users.

Restrictions and guidelines

Make sure the configured measurement unit is the same as the user data measurement unit on the RADIUS server.

Procedure

1. Enter system view.

   `system-view`

2. Enter RADIUS scheme view.

   `radius scheme` *`radius-scheme-name`*

3. Set the data measurement unit for the Remanent_Volume attribute.

   `attribute remanent-volume unit` { **`byte`** | **`giga-byte`** | **`kilo-byte`** | **`mega-byte`** }

   By default, the data measurement unit is kilobyte.

Interpreting the Microsegment-Id attribute to an authorization VLAN

About this task

Use this feature only when the RADIUS server uses authorization microsegment IDs for granular user access control and the access device uses authorization VLANs to implement microsegment-based access control.

This feature enables the device to interpret the RADIUS Microsegment-Id attribute (attribute 182 with vendor ID 25506) assigned by the RADIUS server to an authorization VLAN.

- If the attribute value is an integer, the device interprets this attribute to a VLAN ID.
- If the attribute value is not an integer, the device interprets this attribute to a VLAN name.

Restrictions and guidelines

If the RADIUS server uses a RADIUS attribute other than the Microsegment-Id attribute to assign microsegment IDs, you must first convert the attribute to the Microsegment-Id attribute. To enable RADIUS attribute translation feature, use the **attribute translate** command.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Interpret the Microsegment-Id attribute to an authorization VLAN.

   attribute 182 vendor-id 25506 vlan

   By default, the Microsegment-Id attribute is not interpreted as an authorization VLAN.

Specifying the format for attribute Acct-Session-Id

About this task

RADIUS servers of different types might have different requirements for the format of attribute Acct-Session-Id. The following types are available:

- **Common format**—In this format, the Acct-Session-Id attribute is a string with a minimum length of 38 characters. This string contains the prefix (indicating the access type), date and time, sequence number, LIP address of the access node, device ID, and job ID of the access process.
- **Simplified format**—In this format, the Acct-Session-Id attribute is a string of 16 characters. This string contains the prefix (indicating the access type), month, sequence number, device ID, and LIP address of the access node.

Specify a format for attribute Acct-Session-Id to meet the requirements of the RADIUS servers.

Procedure

1. Enter system view.

   system-view

2. Specify the format for attribute Acct-Session-Id.

   aaa session-id mode { common | simplified }

   By default, the device uses the common format for attribute Acct-Session-Id.

Configuring the RADIUS attribute translation feature

About this task

The RADIUS attribute translation feature enables the device to work correctly with the RADIUS servers of different vendors that support RADIUS attributes incompatible with the device.

RADIUS attribute translation has the following implementations:

- **Attribute conversion**—Converts source RADIUS attributes into destination RADIUS attributes based on RADIUS attribute conversion rules.
- **Attribute rejection**—Rejects RADIUS attributes based on RADIUS attribute rejection rules.

When the RADIUS attribute translation feature is enabled, the device processes RADIUS packets as follows:

- For the sent RADIUS packets:
  - Deletes the rejected attributes from the packets.
  - Uses the destination RADIUS attributes to replace the attributes that match RADIUS attribute conversion rules in the packets.
- For the received RADIUS packets:
  - Ignores the rejected attributes in the packets.
  - Interprets the attributes that match RADIUS attribute conversion rules as the destination RADIUS attributes.

To identify proprietary RADIUS attributes, you can define the attributes as extended RADIUS attributes, and then convert the extended RADIUS attributes to device-supported attributes.

Restrictions and guidelines for RADIUS attribute translation configuration

Configure either conversion rules or rejection rules for a RADIUS attribute.

Configure either direction-based rules or packet type-based rules for a RADIUS attribute.

For direction-based translation of a RADIUS attribute, you can configure a rule for each direction (inbound or outbound). For packet type-based translation of a RADIUS attribute, you can configure a rule for each RADIUS packet type (RADIUS Access-Accept, RADIUS Access-Request, or RADIUS accounting).

Configuring the RADIUS attribute translation feature for a RADIUS scheme

1. Enter system view.

   system-view

2. (Optional.) Define an extended RADIUS attribute.

   radius attribute extended *attribute-name* [**vendor** *vendor-id*] **code** *attribute-code* **type** { **binary** | **date** | **integer** | **interface-id** | **ip** | **ipv6** | **ipv6-prefix** | **octets** | **string** }

3. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

4. Enable the RADIUS attribute translation feature.

   attribute translate

   By default, this feature is disabled.

5. Configure a RADIUS attribute conversion rule or a RADIUS attribute reject rule. Choose the following tasks as needed:
   - Configure a RADIUS attribute conversion rule.

     attribute convert *src-attr-name* **to** *dest-attr-name* { { **access-accept** | **access-request** | **accounting** } * | { **received** | **sent** } * }

By default, no RADIUS attribute conversion rules are configured.

- Configure a RADIUS attribute rejection rule.

  attribute reject *attr-name* { { **access-accept** | **access-request** | **accounting** } * | { **received** | **sent** } * }

  By default, no RADIUS attribute rejection rules are configured.

Configuring the RADIUS attribute translation feature for a RADIUS DAS

1. Enter system view.

   system-view

2. (Optional.) Define an extended RADIUS attribute.

   radius attribute extended *attribute-name* [**vendor** *vendor-id*] **code** *attribute-code* **type** { **binary** | **date** | **integer** | **interface-id** | **ip** | **ipv6** | **ipv6-prefix** | **octets** | **string** }

3. Enter RADIUS DAS view.

   radius dynamic-author server

4. Enable the RADIUS attribute translation feature.

   attribute translate

   By default, this feature is disabled.

5. Configure a RADIUS attribute conversion rule or a RADIUS attribute rejection rule. Choose the following tasks as needed:

   - Configure a RADIUS attribute conversion rule.

     attribute convert *src-attr-name* **to** *dest-attr-name* { { **coa-ack** | **coa-request** } * | { **received** | **sent** } * }

     By default, no RADIUS attribute conversion rules are configured.

   - Configure a RADIUS attribute rejection rule.

     attribute reject *attr-name* { { **coa-ack** | **coa-request** } * | { **received** | **sent** } * }

     By default, no RADIUS attribute rejection rules are configured.

Configuring the RADIUS accounting-on feature

About this task

The accounting-on feature in a RADIUS scheme enables the device to automatically perform the following operations after a reboot:

1. Monitor the status of all accounting servers in the RADIUS scheme.
2. Send accounting-on packets to the reachable servers to request the servers to stop accounting for all online users that use the RADIUS scheme and to log out the users.

You can configure the interval for which the device waits to resend an accounting-on packet and the maximum number of retries.

Procedure

1. Enter system view.

   system-view

2. Enter RADIUS scheme view.

   radius scheme *radius-scheme-name*

3. Enable accounting-on.

   accounting-on enable [**interval** *interval* | **send** *send-times*] *

By default, the accounting-on feature is disabled.

Configuring the RADIUS session-control feature

About this task

Enable this feature for the RADIUS server to dynamically change the user authorization information or forcibly disconnect users by using session-control packets. This task enables the device to receive RADIUS session-control packets on UDP port 1812.

To verify the session-control packets sent from a RADIUS server, specify the RADIUS server as a session-control client to the device.

Restrictions and guidelines

The RADIUS session-control feature can only work with RADIUS servers running on IMC. The session-control client configuration takes effect only when the session-control feature is enabled.

Procedure

1. Enter system view.

   `system-view`

2. Enable the session-control feature.

   `radius session-control enable`

   By default, the session-control feature is disabled.

3. Specify a session-control client.

   `radius session-control client` `{` **`ip`** *`ipv4-address`* `|` **`ipv6`** *`ipv6-address`* `}` `[` **`key`** `{` **`cipher`** `|` **`simple`** `}` *`string`* `|` **`vpn-instance`** *`vpn-instance-name`* `] *`

   By default, no session-control clients are specified.

Configuring the RADIUS DAS feature

About this task

Dynamic Authorization Extensions (DAE) to RADIUS, defined in RFC 5176, can log off online users and change online user authorization information.

In a RADIUS network, the RADIUS server typically acts as the DAE client (DAC) and the NAS acts as the DAE server (DAS).

DAE defines the following types of packets:

- **Disconnect Messages (DMs)**—The DAC sends DM requests to the DAS to log off specific online users.
- **Change of Authorization Messages (CoA Messages)**—The DAC sends CoA requests to the DAS to change the authorization information of specific online users.

When the RADIUS DAS feature is enabled, the NAS performs the following operations:

1. Listens to the default or specified UDP port to receive DAE requests.
2. Logs off online users that match the criteria in the requests and changes their authorization information.
3. Sends DAE responses to the DAC.

Procedure

1. Enter system view.

   `system-view`

2. Enable the RADIUS DAS feature and enter RADIUS DAS view.

```
radius dynamic-author server
```

By default, the RADIUS DAS feature is disabled.

3. Specify a RADIUS DAC.

```
client { ip ipv4-address | ipv6 ipv6-address } [ key { cipher | simple }
string | vendor-id 2011 version { 1.0 | 1.1 } | vpn-instance
vpn-instance-name ] *
```

By default, no RADIUS DACs are specified.

4. (Optional.) Specify the RADIUS DAS port.

```
port port-number
```

By default, the RADIUS DAS port is 3799.

Enabling SNMP notifications for RADIUS

About this task

When SNMP notifications are enabled for RADIUS, the SNMP agent supports the following notifications generated by RADIUS:

- **RADIUS server unreachable notification**—The RADIUS server cannot be reached. RADIUS generates this notification if it does not receive a response to an accounting or authentication request within the specified number of RADIUS request transmission attempts.
- **RADIUS server reachable notification**—The RADIUS server can be reached. RADIUS generates this notification for a previously blocked RADIUS server after the quiet timer expires.
- **Excessive authentication failures notification**—The number of authentication failures compared to the total number of authentication attempts exceeds the specified threshold.

For RADIUS SNMP notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enable SNMP notifications for RADIUS.

```
snmp-agent trap enable radius [ accounting-server-down |
accounting-server-up | authentication-error-threshold |
authentication-server-down | authentication-server-up ] *
```

By default, all SNMP notifications are disabled for RADIUS.

Display and maintenance commands for RADIUS

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the RADIUS scheme configuration.	**display radius scheme** [*radius-scheme-name*]
Display RADIUS packet statistics.	**display radius statistics**
Clear RADIUS statistics.	**reset radius statistics**

Configuring HWTACACS

HWTACACS tasks at a glance

To configure HWTACACS, perform the following tasks:

1. Creating an HWTACACS scheme
2. Specifying the HWTACACS authentication servers
3. Specifying the HWTACACS authorization servers
4. Specifying the HWTACACS accounting servers
5. Specifying the shared keys for secure HWTACACS communication

   Perform this task if no shared keys are specified when configuring HWTACACS servers.
6. Specifying an MPLS L3VPN instance for the scheme

   Perform this task if no MPLS L3VPN instances are specified when configuring HWTACACS servers.
7. (Optional.) Setting HWTACACS timers
8. (Optional.) Configuring parameters for HWTACACS packets
   - Specifying the source IP address for outgoing HWTACACS packets
   - Setting the username format and traffic statistics units
9. (Optional.) Associating an HWTACACS server with a track entry
10. (Optional.) Specifying the action to take for AAA requests if all HWTACACS servers are blocked

Creating an HWTACACS scheme

Restrictions and guidelines

You can configure a maximum of 16 HWTACACS schemes. An HWTACACS scheme can be used by multiple ISP domains.

Procedure

1. Enter system view.

   system-view
2. Create an HWTACACS scheme and enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

Specifying the HWTACACS authentication servers

About this task

You can specify one primary authentication server and a maximum of 16 secondary authentication servers for an HWTACACS scheme. When the primary server is unreachable, the device searches for the secondary servers in the order they are configured. The first secondary server in active state is used for communication.

Restrictions and guidelines

If redundancy is not required, specify only the primary server.

An HWTACACS server can function as the primary authentication server in one scheme and as the secondary authentication server in another scheme at the same time.

Two HWTACACS authentication servers in a scheme, primary or secondary, cannot have the same combination of IP address, VPN instance, and port number.

Procedure

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify the primary HWTACACS authentication server.

   primary authentication { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no primary HWTACACS authentication server is specified.

4. (Optional.) Specify a secondary HWTACACS authentication server.

   secondary authentication { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no secondary HWTACACS authentication servers are specified.

Specifying the HWTACACS authorization servers

About this task

You can specify one primary authorization server and a maximum of 16 secondary authorization servers for an HWTACACS scheme. When the primary server is not available, the device searches for the secondary servers in the order they are configured. The first secondary server in active state is used for communication.

Restrictions and guidelines

If redundancy is not required, specify only the primary server.

An HWTACACS server can function as the primary authorization server of one scheme and as the secondary authorization server of another scheme at the same time.

Two HWTACACS authorization servers in a scheme, primary or secondary, cannot have the same combination of IP address, VPN instance, and port number.

Procedure

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify the primary HWTACACS authorization server.

   primary authorization { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no primary HWTACACS authorization server is specified.

4. (Optional.) Specify a secondary HWTACACS authorization server.

   secondary authorization { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no secondary HWTACACS authorization servers are specified.

Specifying the HWTACACS accounting servers

About this task

You can specify one primary accounting server and a maximum of 16 secondary accounting servers for an HWTACACS scheme. When the primary server is not available, the device searches for the secondary servers in the order they are configured. The first secondary server in active state is used for communication.

Restrictions and guidelines

If redundancy is not required, specify only the primary server.

An HWTACACS server can function as the primary accounting server of one scheme and as the secondary accounting server of another scheme at the same time.

Two HWTACACS accounting servers in a scheme, primary or secondary, cannot have the same combination of IP address, VPN instance, and port number.

HWTACACS does not support accounting for FTP, SFTP, and SCP users.

Procedure

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify the primary HWTACACS accounting server.

   primary accounting { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no primary HWTACACS accounting server is specified.

4. (Optional.) Specify a secondary HWTACACS accounting server.

   secondary accounting { *ipv4-address* | **ipv6** *ipv6-address* } [*port-number* | **key** { **cipher** | **simple** } *string* | **single-connection** | **vpn-instance** *vpn-instance-name*] *

   By default, no secondary HWTACACS accounting servers are specified.

Specifying the shared keys for secure HWTACACS communication

About this task

The HWTACACS client and server use the MD5 algorithm and shared keys to generate the Authenticator value for packet authentication and user password encryption. The client and server must use the same key for each type of communication.

Perform this task to configure shared keys for servers in an HWTACACS scheme. The keys take effect on all servers for which a shared key is not individually configured.

Restrictions and guidelines

Make sure the shared key configured on the device is the same as the shared key configured on the HWTACACS server.

Procedure

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify a shared key for secure HWTACACS authentication, authorization, or accounting communication.

   key { **accounting** | **authentication** | **authorization** } { **cipher** | **simple** } *string*

   By default, no shared key is specified for secure HWTACACS communication.

Specifying an MPLS L3VPN instance for the scheme

About this task

The VPN instance specified for an HWTACACS scheme applies to all servers in that scheme. If a VPN instance is also configured for an individual HWTACACS server, the VPN instance specified for the HWTACACS scheme does not take effect on that server.

Procedure

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify a VPN instance for the HWTACACS scheme.

   vpn-instance *vpn-instance-name*

   By default, an HWTACACS scheme belongs to the public network.

Setting HWTACACS timers

About this task

The device uses the following timers to control communication with an HWTACACS server:

- **Server response timeout timer (response-timeout)**—Defines the HWTACACS server response timeout timer. The device starts this timer immediately after an HWTACACS authentication, authorization, or accounting request is sent. If the device does not receive a response from the server within the timer, it sets the server to blocked. Then, the device sends the request to another HWTACACS server.
- **Real-time accounting timer (realtime-accounting)**—Defines the interval at which the device sends real-time accounting packets to the HWTACACS accounting server for online users.
- **Server quiet timer (quiet)**—Defines the duration to keep an unreachable server in blocked state. If a server is not reachable, the device changes the server status to blocked, starts this timer for the server, and tries to communicate with another server in active state. After the server quiet timer expires, the device changes the status of the server back to active.

The server quiet timer setting affects the status of HWTACACS servers. If the scheme includes one primary HWTACACS server and multiple secondary HWTACACS servers, the device communicates with the HWTACACS servers based on the following rules:

- When the primary server is in active state, the device communicates with the primary server. When the primary server is unreachable, the device researches a secondary server in active status in the order they are configured.
- When one or more servers are in active state, the device tries to communicate with these servers only, even if they are unreachable.
- When all servers are in blocked state, the device only tries to communicate with the primary server.

- If the primary server is unreachable, the device changes the server status to blocked and starts a quiet timer for the server. When the quiet timer of the server expires, the status of the server changes back to active. The device does not check the server again during the authentication, authorization, or accounting process.
- The search process continues until the device finds an available secondary server or has checked all secondary servers in active state. If no server is available, the device considers the authentication, authorization, or accounting attempt a failure.
- When you remove a server in use, communication with the server times out. The device looks for a server in active state by first checking the primary server, and then checking secondary servers in the order they are configured.
- When an HWTACACS server's status changes automatically, the device changes this server's status accordingly in all HWTACACS schemes in which this server is specified.

Restrictions and guidelines

A short real-time accounting interval helps improve accounting precision but requires many system resources. When there are 1000 or more users, set a real-time accounting interval longer than 15 minutes.

Procedure

1. Enter system view.

   `system-view`

2. Enter HWTACACS scheme view.

   `hwtacacs scheme` *`hwtacacs-scheme-name`*

3. Set the HWTACACS timers. Choose the following tasks as needed:
   - Set the HWTACACS server response timeout timer.

     `timer response-timeout` *`seconds`*

     By default, the HWTACACS server response timeout timer is 5 seconds.
   - Set the real-time accounting interval.

     `timer realtime-accounting` *`minutes`*

     By default, the real-time accounting interval is 12 minutes.
   - Set the server quiet timer.

     `timer quiet` *`minutes`*

     By default, the server quiet timer is 5 minutes.

Specifying the source IP address for outgoing HWTACACS packets

About this task

The source IP address of HWTACACS packets that a NAS sends must match the IP address of the NAS configured on the HWTACACS server. An HWTACACS server identifies a NAS by IP address. When the HWTACACS server receives a packet, it checks the source IP address of the packet.

- If it is the IP address of a managed NAS, the server processes the packet.
- If it is not the IP address of a managed NAS, the server drops the packet.

Before sending an HWTACACS packet, the NAS selects a source IP address in the following order:

1. The source IP address specified for the HWTACACS scheme.
2. The source IP address specified in system view for the VPN or public network, depending on where the HWTACACS server resides.
3. The IP address of the outbound interface specified by the route.

Restrictions and guidelines for source IP address configuration

You can specify the source IP address for outgoing HWTACACS packets in HWTACACS scheme view or in system view.

- The IP address specified in HWTACACS scheme view applies to one HWTACACS scheme.
- The IP address specified in system view applies to all HWTACACS schemes.

The source IP address of HWTACACS packets that a NAS sends must match the IP address of the NAS that is configured on the HWTACACS server.

As a best practice, specify a loopback interface address as the source IP address for outgoing HWTACACS packets to avoid HWTACACS packet loss caused by physical port errors.

To communicate with the HWTACACS server, the source address of outgoing HWTACACS packets is typically the IP address of an egress interface on the NAS.

Specifying a source IP address for all HWTACACS schemes

1. Enter system view.

   system-view

2. Specify a source IP address for outgoing HWTACACS packets.

   hwtacacs nas-ip { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]

   By default, the source IP address of an HWTACACS packet sent to the server is the primary IPv4 address or the IPv6 address of the outbound interface.

Specifying a source IP address for an HWTACACS scheme

1. Enter system view.

   system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Specify a source IP address for outgoing HWTACACS packets.

   nas-ip { *ipv4-address* | **ipv6** *ipv6-address* }

   By default, the source IP address of an outgoing HWTACACS packet is that configured by using the **hwtacacs nas-ip** command in system view. If the **hwtacacs nas-ip** command is not used, the source IP address is the primary IP address of the outbound interface.

Setting the username format and traffic statistics units

About this task

A username is typically in the *userid@isp-name* format, where the *isp-name* part represents the user's ISP domain name. By default, the ISP domain name is included in a username. If HWTACACS servers do not recognize usernames that contain ISP domain names, you can configure the device to send usernames without domain names to the servers.

The device reports online user traffic statistics in accounting packets.

Restrictions and guidelines

If two or more ISP domains use the same HWTACACS scheme, configure the HWTACACS scheme to keep the ISP domain name in usernames for domain identification.

For accounting accuracy, make sure the traffic measurement units configured on the device are the same as the traffic measurement units configured on the HWTACACS accounting servers.

Procedure

1. Enter system view.

system-view

2. Enter HWTACACS scheme view.

   hwtacacs scheme *hwtacacs-scheme-name*

3. Set the format of usernames sent to the HWTACACS servers.

   user-name-format { **keep-original** | **with-domain** | **without-domain** }

   By default, the ISP domain name is included in a username.

4. Set the data flow and packet measurement units for traffic statistics.

   data-flow-format { **data** { **byte** | **giga-byte** | **kilo-byte** | **mega-byte** } | **packet** { **giga-packet** | **kilo-packet** | **mega-packet** | **one-packet** } }*

   By default, traffic is counted in bytes and packets.

Associating an HWTACACS server with a track entry

About this task

Perform this task on a network that has high real-time requirements for HWTACACS authentication, authorization, and accounting.

By default, the device does not actively detect the status of an HWTACACS server. It changes the state of an HWTACACS server to active or blocked based on the server response timeout timer and the server quiet timer. This timer-based state transition mechanism needs time to determine the server state, and it cannot ensure that the device obtains the actual server state in time. To resolve this issue, associate the server with a track entry and associate the track entry with a TCP-type NQA operation. This HWTACACS server-Track-NQA collaboration can actively detect the reachability of the server in real time.

By using HWTACACS server-Track-NQA collaboration, the device determines the status of an HWTACACS server only based on the detection result.

1. The NQA operation starts to detect the reachability of the server and obtains the result. NQA sends the detection result to the Track module for the Track module to set the state of the track entry.
   - If the server is reachable, the Track module sets the state of the track entry to Positive.
   - If the server is unreachable, the Track module sets the state of the track entry to Negative.
   - If the Track-NQA collaboration does not take effect, the Track module keeps the track entry in NotReady state or changes its state to NotReady.
2. AAA sets the status of the server based on the track entry state.
   - If the track entry is in Positive state, AAA sets the state of the server to active.
   - If the track entry is in Negative state, AAA sets the state of the server to blocked and disables the quiet timer for the server.
   - If the track entry stays in NotReady state or its state changes to NotReady, AAA sets the state of the server to active.

Prerequisites

Before you perform this task, you must complete the following tasks:

- Configure an NQA operation of the TCP type and start the NQA operation. For more information, see NQA configuration in *High Availability and Monitoring Configuration Guide*.
- Configure a track entry associated with the NQA operation. For more information, see Track configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Associate an HWTACACS server with a track entry to detect the server reachability.

   ```
   hwtacacs server-probe { ip ipv4-address | ipv6 ipv6-address }
   [ vpn-instance vpn-instance-name ] [ port port-number ] track
   track-entry-number
   ```

   By default, an HWTACACS server is not associated with a track entry.

Specifying the action to take for AAA requests if all HWTACACS servers are blocked

About this task

If all servers in an HWTACACS scheme are blocked, the device takes one of the following actions upon receiving AAA requests in the domain that uses the scheme:

- **attempt**—Attempts to connect to the server that has the highest priority in the scheme. (Typically, the highest-priority server is the primary server. If no primary server is specified, it is the firstly configured secondary server.) If the device fails to connect to the server, it turns to the backup method.
- **skip**—Skips all servers in the scheme and turns to the backup method.

The **attempt** action gives the device a chance to use the scheme in case the server with the highest priority in the scheme might be available. However, the attempt to communicate with an unavailable server increases the response time for AAA requests. As a best practice, specify the **skip** action in scenarios that require quick responses to AAA requests.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter HWTACACS scheme view.

   ```
   hwtacacs scheme hwtacacs-scheme-name
   ```

3. Specify the action to take for AAA requests if all servers in the scheme are blocked.

   ```
   server-block-action { attempt | skip }
   ```

   By default, the **attempt** action applies.

Display and maintenance commands for HWTACACS

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the configuration or server statistics of HWTACACS schemes.	**display hwtacacs scheme** [*hwtacacs-scheme-name* [**statistics**]]
Clear HWTACACS statistics.	**reset hwtacacs statistics** { **accounting** | **all** | **authentication** | **authorization** }

Configuring LDAP

LDAP tasks at a glance

To configure LDAP, perform the following tasks:

1. Configuring an LDAP server
    a. Creating an LDAP server
    b. Configuring the IP address of the LDAP server
    c. (Optional.) Specifying the LDAP version
    d. (Optional.) Specifying the source IP address of outgoing LDAP packets
    e. (Optional.) Setting the LDAP server timeout period
    f. Configuring administrator attributes
    g. Configuring LDAP user attributes
    h. Specifying the character encoding format
    i. (Optional.) Configuring a user group filter
2. Configuring an LDAP attribute map
3. Creating an LDAP scheme
4. Specifying the LDAP authentication server
5. (Optional.) Specifying the LDAP authorization server
6. (Optional.) Specifying an LDAP attribute map for LDAP authorization

Creating an LDAP server

1. Enter system view.
   `system-view`
2. Create an LDAP server and enter LDAP server view.
   `ldap server server-name`

Configuring the IP address of the LDAP server

Restrictions and guidelines

You can configure an IPv4 address or an IPv6 address for an LDAP server. If you configure the IP address for an LDAP server multiple times, the most recent configuration takes effect.

Procedure

1. Enter system view.
   `system-view`
2. Enter LDAP server view.
   `ldap server server-name`
3. Configure the IP address of the LDAP server.
   ```
   { ip ipv4-address | ipv6 ipv6-address } [ port port-number ]
   [ vpn-instance vpn-instance-name ]
   ```
   By default, an LDAP server does not have an IP address.

Specifying the LDAP version

Restrictions and guidelines

The device supports LDAPv2 and LDAPv3.

A Microsoft LDAP server supports only LDAPv3.

The LDAP version specified on the device must be consistent with the version specified on the LDAP server.

Procedure

1. Enter system view.

   system-view

2. Enter LDAP server view.

   ldap server *server-name*

3. Specify the LDAP version.

   protocol-version { **v2** | **v3** }

   By default, LDAPv3 is used.

Specifying the source IP address of outgoing LDAP packets

About this task

Perform this task to specify a source IP address for LDAP packets sent to an LDAP server.

Restrictions and guidelines

When you perform this task for an LDAP server, follow these restrictions and guidelines:

- You can specify only one source IPv4 address and one source IPv6 address for the LDAP packets sent to the server.
- You can specify only one source interface. Make sure the source interface can reach the server.
- To have the source interface configuration take effect, make sure the source interface is in the same VPN instance as the server.
- To have the source address configuration take effect, the IP version of the specified source address must be the same as that of the server IP address.
- The source interface configuration and the source IP address configuration overwrite each other.

Procedure

4. Enter system view.

   system-view

5. Enter LDAP server view.

   ldap server *server-name*

6. Specify a source IP address for LDAP packets sent to the LDAP server.

   source-ip { *ipv4-address* | **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }

   By default, no IP address is specified as the source IP address of LDAP packets sent to an LDAP server. The device uses the primary IPv4 address or the IPv6 address of the outbound interface that can reach the server as the source IP address of LDAP packets sent to the server.

Setting the LDAP server timeout period

About this task

If the device sends a bind or search request to an LDAP server without receiving the server's response within the server timeout period, the authentication or authorization request times out. Then, the device tries the backup authentication or authorization method. If no backup method is configured in the ISP domain, the device considers the authentication or authorization attempt a failure.

Procedure

1. Enter system view.

   system-view

2. Enter LDAP server view.

   ldap server *server-name*

3. Set the LDAP server timeout period.

   server-timeout *time-interval*

   By default, the LDAP server timeout period is 10 seconds.

Configuring administrator attributes

About this task

To configure the administrator DN and password for binding with the LDAP server during LDAP authentication:

Procedure

1. Enter system view.

   system-view

2. Enter LDAP server view.

   ldap server *server-name*

3. Specify the administrator DN.

   login-dn *dn-string*

   By default, no administrator DN is specified.

   The administrator DN specified on the device must be the same as the administrator DN configured on the LDAP server.

4. Configure the administrator password.

   login-password { **cipher** | **simple** } *string*

   By default, no administrator password is specified.

Configuring LDAP user attributes

About this task

To authenticate a user, an LDAP client must complete the following operations:

1. Establish a connection to the LDAP server.
2. Obtain the user DN from the LDAP server.
3. Use the user DN and the user's password to bind with the LDAP server.

LDAP provides a DN search mechanism for obtaining the user DN. According to the mechanism, an LDAP client sends search requests to the server based on the search policy determined by the LDAP user attributes of the LDAP client.

The LDAP user attributes include:

- Search base DN.
- Search scope.
- Username attribute.
- Username format.
- User object class.

Restrictions and guidelines

If the LDAP server contains many directory levels, a user DN search starting from the root directory can take a long time. To improve efficiency, you can change the start point by specifying the search base DN.

Procedure

1. Enter system view.

   `system-view`

2. Enter LDAP server view.

   `ldap server` *`server-name`*

3. Specify the user search base DN.

   `search-base-dn` *`base-dn`*

   By default, no user search base DN is specified.

4. (Optional.) Specify the user search scope.

   `search-scope` { **`all-level`** | **`single-level`** }

   By default, the user search scope is **`all-level`**.

5. (Optional.) Specify the username attribute.

   `user-parameters user-name-attribute` { *`name-attribute`* | **`cn`** | **`uid`** }

   By default, the username attribute is **cn**.

6. (Optional.) Specify the username format.

   `user-parameters user-name-format` { **`with-domain`** | **`without-domain`** }

   By default, the username format is **`without-domain`**.

7. (Optional.) Specify the user object class.

   `user-parameters user-object-class` *`object-class-name`*

   By default, no user object class is specified, and the default user object class on the LDAP server is used. The default user object class for this command varies by server model.

Specifying the character encoding format

About this task

By default, the device encodes the configuration made through the Web interface in GB18030 and that made through terminal software in the character encoding format used by the software. If the device and the LDAP server use different character encoding formats, some characters in the information exchanged between them might fail to be interpreted, causing further issues. For example, if user DNs on the LDPA server are Chinese and the user DNs on the device are English, user DN search will fail and then the users will fail to come online. To resolve this issue, use this feature to ensure that the device and the LDAP server use the same character encoding format during information exchange.

After you specify the character encoding format for an LDAP server, the device processes LDAP packets exchanged with the LDAP server as follows:

- For an LDAP packet sent to the LDAP server, the device first decodes the information in the packet by using GB18030. Then, the device uses the specified character encoding format to encode the information.
- For an LDAP packet received form the LDAP server, the device first uses the specified character encoding format to decode the information in the packet. Then, the device uses GB18030 to encode the information and saves the information.

Restriction and guidelines

As a best practice to avoid LDAP authentication failure caused by inconsistent character encoding format, change the character encoding format before using the LDAP server to perform authentication on users.

Procedure

1. Enter system view.

   system-view

2. Enter LDAP server view.

   ldap server *server-name*

3. Specify the character encoding format for the LDAP server.

   character-encoding { **gb18030** | **utf-8** }

   By default, no character encoding format is specified for an LDAP server. The device does not change the character encoding format for information exchanged with the LDAP server.

Configuring a user group filter

About this task

When the device requests to import user group information from an LDAP server, the LDAP server sends only user groups that match the user group filter to the device.

Procedure

1. Enter system view.

   system-view

2. Enter LDAP server view.

   ldap server *server-name*

3. Configure the user group filter.

   group-filter *group-filter*

   By default, the user group filter is **(objectclass=group)**.

Configuring an LDAP attribute map

About this task

Configure an LDAP attribute map to define a list of LDAP-AAA attribute mapping entries. To apply the LDAP attribute map, specify the name of the LDAP attribute map in the LDAP scheme used for authorization.

The LDAP attribute map feature enables the device to convert LDAP attributes obtained from an LDAP authorization server to device-recognizable AAA attributes based on the mapping entries. Because the device ignores unrecognized LDAP attributes, configure the mapping entries to include important LDAP attributes that should not be ignored.

An LDAP attribute can be mapped only to one AAA attribute. Different LDAP attributes can be mapped to the same AAA attribute.

Procedure

1. Enter system view.

   system-view

2. Create an LDAP attribute map and enter LDAP attribute map view.

   ldap attribute-map *map-name*

3. Configure a mapping entry.

   map ldap-attribute *ldap-attribute-name* [**prefix** *prefix-value* **delimiter** *delimiter-value*] **aaa-attribute user-group**

Creating an LDAP scheme

Restrictions and guidelines

You can configure a maximum of 16 LDAP schemes. An LDAP scheme can be used by multiple ISP domains.

Procedure

1. Enter system view.

   system-view

2. Create an LDAP scheme and enter LDAP scheme view.

   ldap scheme *ldap-scheme-name*

Specifying the LDAP authentication server

1. Enter system view.

   system-view

2. Enter LDAP scheme view.

   ldap scheme *ldap-scheme-name*

3. Specify the LDAP authentication server.

   authentication-server *server-name*

   By default, no LDAP authentication server is specified.

Specifying the LDAP authorization server

1. Enter system view.

   system-view

2. Enter LDAP scheme view.

   ldap scheme *ldap-scheme-name*

3. Specify the LDAP authorization server.

   authorization-server *server-name*

   By default, no LDAP authorization server is specified.

Specifying an LDAP attribute map for LDAP authorization

About this task

Specify an LDAP attribute map for LDAP authorization to convert LDAP attributes obtained from the LDAP authorization server to device-recognizable AAA attributes.

Restrictions and guidelines

You can specify only one LDAP attribute map in an LDAP scheme.

Procedure

1. Enter system view.

```
system-view
```

2. Enter LDAP scheme view.

```
ldap scheme ldap-scheme-name
```

3. Specify an LDAP attribute map.

```
attribute-map map-name
```

By default, no LDAP attribute map is specified.

Display and maintenance commands for LDAP

Execute **display** commands in any view.

Task	Command
Display the configuration of LDAP schemes.	**display ldap scheme** [*ldap-scheme-name*]

Creating an ISP domain

About ISP domains

In a networking scenario with multiple ISPs, the device can connect to users of different ISPs. These users can have different user attributes, such as different username and password structures, different service types, and different rights. To manage users of different ISPs, configure authentication, authorization, and accounting methods and domain attributes for each ISP domain as needed.

The device supports a maximum of 1024 ISP domains, including the system-defined ISP domain **system**. You can specify one of the ISP domains as the default domain.

On the device, each user belongs to an ISP domain. If a user does not provide an ISP domain name at login, the device considers the user belongs to the default ISP domain.

Each ISP domain has a set of system-defined AAA methods, which are local authentication, local authorization, and local accounting. If you do not configure any AAA methods for an ISP domain, the device uses the system-defined AAA methods for users in the domain.

The device chooses an authentication domain for each user in the following order:

1. The authentication domain specified for the access module.
2. The ISP domain in the username.
3. The default ISP domain of the device.

If the chosen domain does not exist on the device, the device searches for the ISP domain that accommodates users assigned to nonexistent domains. (Support for the authentication domain configuration depends on the access module.) If no such ISP domain is configured, user authentication fails.

Restrictions and guidelines for ISP domain configuration

An ISP domain cannot be deleted when it is the default ISP domain. Before you use the **undo domain** command, change the domain to a non-default ISP domain by using the **undo domain default enable** command.

You can modify the settings of the system-defined ISP domain **system**, but you cannot delete the domain.

To avoid RADIUS authentication, authorization, or accounting failures, use short domain names to ensure that usernames containing a domain name do not exceed 253 characters.

To avoid RADIUS accounting failures, make sure the domain name contained in usernames sent to the RADIUS server does not exceed 247 characters.

Creating an ISP domain

1. Enter system view.

   ```
   system-view
   ```

2. Create an ISP domain and enter ISP domain view.

   ```
   domain isp-name
   ```

   By default, a system-defined ISP domain exists. The domain name is **system**.

Specifying the default ISP domain

1. Enter system view.

   ```
   system-view
   ```

2. Specify the default ISP domain.

   ```
   domain default enable isp-name
   ```

   By default, the default ISP domain is the system-defined ISP domain **system**.

Specifying an ISP domain for users that are assigned to nonexistent domains

1. Enter system view.

   ```
   system-view
   ```

2. Specify the ISP domain to accommodate users that are assigned to nonexistent domains.

   ```
   domain if-unknown isp-name
   ```

   By default, no ISP domain is specified to accommodate users that are assigned to nonexistent domains.

Configuring ISP domain attributes

Setting ISP domain status

About this task

By placing the ISP domain in active or blocked state, you allow or deny network service requests from users in the domain.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter ISP domain view.

   ```
   domain isp-name
   ```

3. Set the status of the ISP domain.

   ```
   state { active | block }
   ```

By default, an ISP domain is in active state, and users in the domain can request network services.

Configuring authorization attributes for an ISP domain

About this task

The device supports the following authorization attributes:

- **ACL**—The device restricts authenticated users to access only the network resources permitted by the ACL.
- **CAR action**—The attribute controls the traffic flow of authenticated users.
- **Idle cut**—The device logs out a user if the user's total traffic in the idle timeout period at the specified direction is less than the specified minimum traffic.
- **Maximum number of multicast groups**—The attribute restricts the maximum number of multicast groups that an authenticated user can join concurrently.
- **IPv4 address pool**—The device assigns IPv4 addresses from the pool to authenticated users in the domain.
- **IPv6 address pool**—The device assigns IPv6 addresses from the pool to authenticated users in the domain.
- **IPv6 prefix**—The device authorizes the IPv6 prefix to authenticated users in the domain.
- **DNS server address**—The attribute specifies the DNS server that offers DNS services to authenticated users in the domain.
- **Session timeout time**—The device logs off a user when the user's session timeout timer expires.
- **Redirect URL**—The device redirects users in the domain to the URL after they pass authentication.
- **User group**—Authenticated users in the domain obtain all attributes of the user group.
- **VPN instance**—The device allows authenticated users in the domain to access network resources in the authorization VPN.

The device assigns the authorization attributes (excluding the idle cut attribute) in the ISP domain to the authenticated users that do not receive these attributes from the server.

If the idle cut attribute is configured in an ISP domain, the device assigns the attribute to the authenticated users in the domain. If no idle cut attribute is configured in the ISP domain, the device uses the idle cut attribute assigned by the server.

Procedure

1. Enter system view.

   system-view

2. Enter ISP domain view.

   domain *isp-name*

3. Configure authorization attributes for authenticated users in the ISP domain.

   authorization-attribute { **acl** *acl-number* | **car inbound cir** *committed-information-rate* [**pir** *peak-information-rate*] **outbound cir** *committed-information-rate* [**pir** *peak-information-rate*] | **idle-cut** *minutes* [*flow*] | **igmp max-access-number** *max-access-number* | **ip-pool** *ipv4-pool-name* | **ipv6-pool** *ipv6-pool-name* | **ipv6-prefix** *ipv6-prefix prefix-length* | { **primary-dns** | **secondary-dns** } { **ip** *ipv4-address* | **ipv6** *ipv6-address* } | **session-timeout** *minutes* | **url** *url-string* | **user-group** *user-group-name* | **vpn-instance** *vpn-instance-name* }

   The default settings are as follows:

- The idle cut feature is disabled.
- An IPv4 user can concurrently join a maximum of four IGMP multicast groups.
- No other authorization attributes exist.

Including the idle timeout period in the user online duration to be sent to the server

About this task

If a user goes offline due to connection failure or malfunction, the user's online duration sent to the server includes the idle timeout period assigned by the authorization server. The online duration generated on the server is longer than the actual online duration of the user.

For portal users, the device includes the idle timeout period set for the online portal user detection feature in the user online duration. For more information about online detection for portal users, see "Configuring portal authentication."

Procedure

1. Enter system view.

   `system-view`

2. Enter ISP domain view.

   `domain` *`isp-name`*

3. Configure the device to include the idle timeout period in the user online duration to be sent to the server.

   `session-time include-idle-time`

   By default, the user online duration sent to the server does not include the idle timeout period.

Specifying the user address type in an ISP domain

1. Enter system view.

   `system-view`

2. Enter ISP domain view.

   `domain` *`isp-name`*

3. Specify the user address type in the ISP domain.

   `user-address-type` { **`ds-lite`** | **`ipv6`** | **`nat64`** | **`private-ds`** | **`private-ipv4`** | **`public-ds`** | **`public-ipv4`** }

   By default, no user address type is specified.

Specifying the service type for users in an ISP domain

1. Enter system view.

   `system-view`

2. Enter ISP domain view.

   `domain` *`isp-name`*

3. Specify the service type for users in the ISP domain.

   `service-type` { **`hsi`** | **`stb`** | **`voip`** }

   By default, the service type is **hsi**.

Specifying the types of IP addresses that L2TP users must rely on to use the basic services

About this task

An L2TP user might request multiple services of different IP address types. By default, the device logs off the user if the user does not obtain an IP address. This feature enables the device to allow the user to come online if the user has obtained IP addresses of all the specified types for the basic services.

Restrictions and guidelines

This feature takes effect only when the device acts as an L2TP LNS and only on L2TP users.

Procedure

1. Enter system view.

   system-view

2. Enter ISP domain view.

   domain *isp-name*

3. Specify the types of IP addresses that L2TP users must rely on to use the basic services.

   basic-service-ip-type { **ipv4** | **ipv6** | **ipv6-pd** } *

   By default, L2TP users do not rely on any types of IP addresses to use the basic services.

Setting the IPv6 address wait timer for L2TP users

About this task

The IPv6 address wait timer defines the maximum amount of time that a user can wait before the device determines that the user fails to obtain an IPv6 address or PD prefix.

The device starts an IPv6 address wait timer for an L2TP user after it finishes IPv6CP negotiation with the user. If the user's basic service relies on an IPv6 address or PD prefix but it fails to obtain any IPv6 address or PD prefix when the timer expires, the user cannot come online.

Restrictions and guidelines

This feature takes effect only when the device acts as an L2TP LNS and only on L2TP users.

As a best practice, increase the IPv6 address wait timer in the following situations:

- The network communication is unstable.
- The device uses DHCPv6 to assign IPv6 addresses to users.
- The ISP domain serves a large number of users.

Procedure

1. Enter system view.

   system-view

2. Enter ISP domain view.

   domain *isp-name*

3. Set the IPv6 address wait timer for L2TP users.

   dhcpv6-follow-ipv6cp timeout *delay-time*

   By default, the IPv6 address wait timer for L2TP users is 60 seconds.

Configuring AAA methods for an ISP domain

Configuring authentication methods for an ISP domain

Restrictions and guidelines

When you configure authentication, follow these restrictions and guidelines:

- For high availability, you can specify one primary authentication method and multiple backup authentication methods. When the primary method is invalid, the device attempts to use the backup methods in sequence.

  For example, the **`radius-scheme`** *`radius-scheme-name`* **`local`** **none** parameters specify a primary RADIUS authentication method and two backup methods (local authentication and no authentication). The device performs RADIUS authentication by default and performs local authentication when the RADIUS server is invalid. The device does not perform authentication when both of the previous methods are invalid.

  The remote authentication method is invalid in the following situations:

  - The specified authentication scheme does not exist.
  - Authentication packet sending fails.
  - The device does not receive any authentication response packets from an authentication server.

  The local authentication method is invalid if the device fails to find the matching local user configuration.

- If the authentication method uses a RADIUS scheme and the authorization method does not use a RADIUS scheme, AAA accepts only the authentication result from the RADIUS server. The Access-Accept message from the RADIUS server also includes the authorization information, but the device ignores the information.
- If an HWTACACS scheme is specified, the device uses the entered username for role authentication.
- If a RADIUS scheme is specified, the device uses username **$enab*n*$** on the RADIUS server for role authentication.
  - To obtain a level-*n* user role, you must create a user account for the level-*n* user role in the **$enabn$** format on the RADIUS server. The variable *n* represents the target user role level.
  - To obtain a non-level-*n* user role, you must perform the following tasks:
    - Create a user account named **$enab0$** on the server.
    - Configure the cisco-av-pair attribute for the account in the form of **allowed-roles="***role***"**. The variable *role* represents the target user role.

  For more information about user role authentication, see *Fundamentals Configuration Guide*.

When the primary authentication method is local, the following rules apply to the authentication of a user:

- The device uses the backup authentication methods in sequence only if local authentication is invalid for one of the following reasons:
  - An exception occurs in the local authentication process.
  - The user account is not configured on the device or the user is not allowed to use the access service.
- The device does not turn to the backup authentication methods if local authentication is invalid because of any other reason. Authentication fails for the user.

If you specify multiple authentication methods for SSL VPN users in an ISP domain, the device does not support the online user password change feature for the SSL VPN users.

If you specify an LDAP scheme for SSL VPN users in an ISP domain, the device does not support the online user password change feature for the SSL VPN users.

Prerequisites

Before configuring authentication methods, complete the following tasks:

1. Determine the access type or service type to be configured. With AAA, you can configure an authentication method for each access type and service type.
2. Determine whether to configure the default authentication method for all access types or service types. The default authentication method applies to all access users. However, the method has a lower priority than the authentication method that is specified for an access type or service type.

Procedure

1. Enter system view.

   system-view

2. Enter ISP domain view.

   domain *isp-name*

3. (Optional.) Specify default authentication methods for all types of users.

   authentication default { **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **ldap-scheme** *ldap-scheme-name* [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

   By default, the default authentication method is **local**.

4. Specify authentication methods for a user type or a service.
   - Specify authentication methods for ADVPN users.

     authentication advpn { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authentication methods are used for ADVPN users.
   - Specify authentication methods for IPoE users.

     authentication ipoe { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authentication methods are used for IPoE users.
   - Specify extended authentication methods for IKE users.

     authentication ike { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authentication methods are used for IKE extended authentication.
   - Specify authentication methods for login users.

     authentication login { **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **ldap-scheme** *ldap-scheme-name* [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] | **local** [**ldap-scheme** *ldap-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

     By default, the default authentication methods are used for login users.
   - Specify authentication methods for portal users.

     authentication portal { **ldap-scheme** *ldap-scheme-name* [**local**] [**none**] | **local** [**ldap-scheme** *ldap-scheme-name* | **radius-scheme**

```
radius-scheme-name ] [ none ] | none | radius-scheme
radius-scheme-name [ local ] [ none ] }
```

By default, the default authentication methods are used for portal users.

- Specify authentication methods for PPP users.

```
authentication ppp { hwtacacs-scheme hwtacacs-scheme-name
[ radius-scheme radius-scheme-name ] [ local ] [ none ] | ldap-scheme
ldap-scheme-name [ local ] [ none ] | local [ radius-scheme
radius-scheme-name | hwtacacs-scheme hwtacacs-scheme-name ] *
[ none ] local [ ldap-scheme ldap-scheme-name ] [ none ] | | none |
radius-scheme radius-scheme-name [ hwtacacs-scheme
hwtacacs-scheme-name ] [ local ] [ none ] }
```

By default, the default authentication methods are used for PPP users.

- Specify authentication methods for SSL VPN users.

```
authentication sslvpn { ldap-scheme ldap-scheme-name [ local ]
[ none ] | local [ ldap-scheme ldap-scheme-name | radius-scheme
radius-scheme-name ] [ none ] | none | radius-scheme
radius-scheme-name [ local ] [ none ] }
```

By default, the default authentication methods are used for SSL VPN users.

- Specify authentication methods for obtaining a temporary user role.

```
authentication super { hwtacacs-scheme hwtacacs-scheme-name |
radius-scheme radius-scheme-name } *
```

By default, the default authentication methods are used for obtaining a temporary user role.

Configuring authorization methods for an ISP domain

Restrictions and guidelines

For high availability, you can specify one primary authorization method and multiple backup authorization methods. When the primary method is invalid, the device attempts to use the backup methods in sequence. For example, the **radius-scheme** *radius-scheme-name* **local none** parameters specify a primary RADIUS authorization method and two backup methods (local authorization and no authorization). The device performs RADIUS authorization by default and performs local authorization when the RADIUS server is invalid. The device does not perform authorization when both of the previous methods are invalid.

The remote authorization method is invalid in the following situations:

- The specified authorization scheme does not exist.
- Authorization packet sending fails.
- The device does not receive any authorization response packets from an authorization server.

The local authorization method is invalid if the device fails to find the matching local user configuration.

Only SSL VPN users support LDAP authorization in the current software version.

To use a RADIUS scheme as the authorization method, specify the name of the RADIUS scheme that is configured as the authentication method for the ISP domain. If an invalid RADIUS scheme is specified as the authorization method, RADIUS authentication and authorization fail.

When the primary authorization method is local, the following rules apply to the authorization of a user:

- The device uses the backup authorization methods in sequence only if local authorization is invalid for one of the following reasons:
    - An exception occurs in the local authorization process.

  - The user account is not configured on the device or the user is not allowed to use the access service.
- The device does not turn to the backup authorization methods if local authorization is invalid because of any other reason. Authorization fails for the user.

Prerequisites

Before configuring authorization methods, complete the following tasks:

1. Determine the access type or service type to be configured. With AAA, you can configure an authorization scheme for each access type and service type.
2. Determine whether to configure the default authorization method for all access types or service types. The default authorization method applies to all access users. However, the method has a lower priority than the authorization method that is specified for an access type or service type.

Procedure

1. Enter system view.

   system-view

2. Enter ISP domain view.

   domain *isp-name*

3. (Optional.) Specify default authorization methods for all types of users.

   authorization default { **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

   By default, the authorization method is **local**.

4. Specify authorization methods for a user type or a service.
   - Specify authorization methods for ADVPN users.

     authorization advpn { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authorization methods are used for ADVPN users.
   - Specify command authorization methods.

     authorization command { **hwtacacs-scheme** *hwtacacs-scheme-name* [**local**] [**none**] | **local** [**none**] | **none** }

     By default, the default authorization methods are used for command authorization.
   - Specify authorization methods for IKE extended authentication.

     authorization ike { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authorization methods are used for IKE extended authentication.
   - Specify authorization methods for IPoE users.

     authorization ipoe { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

     By default, the default authorization methods are used for IPoE users.
   - Specify authorization methods for login users.

     authorization login { **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] | **none** | **radius-scheme**

radius-scheme-name [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

By default, the default authorization methods are used for login users.

- Specify authorization methods for portal users.

  authorization portal { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

  By default, the default authorization methods are used for portal users.

- Specify authorization methods for PPP users.

  authorization ppp { **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **ldap-scheme** *ldap-scheme-name* [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] **local** [**ldap-scheme** *ldap-scheme-name*] [**none**] | | **none** | **radius-scheme** *radius-scheme-name* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

  By default, the default authorization methods are used for PPP users.

- Specify authorization methods for SSL VPN users.

  authorization sslvpn { **ldap-scheme** *ldap-scheme-name* [**local**] [**none**] | **local** [**ldap-scheme** *ldap-scheme-name* | **radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

  By default, the default authorization methods are used for SSL VPN users.

Configuring accounting methods for an ISP domain

Restrictions and guidelines

For high availability, you can specify one primary accounting method and multiple backup accounting methods. When the primary method is invalid, the device attempts to use the backup methods in sequence. For example, the **radius-scheme** *radius-scheme-name* **local none** parameters specify a primary RADIUS accounting method and two backup methods (local accounting and no accounting). The device performs RADIUS accounting by default and performs local accounting when the RADIUS server is invalid. The device does not perform accounting when both of the previous methods are invalid.

The remote accounting method is invalid in the following situations:

- The specified accounting scheme does not exist.
- Accounting packet sending fails.
- The device does not receive any accounting response packets from an accounting server.

The local accounting method is invalid if the device fails to find the matching local user configuration.

FTP, SFTP, and SCP users do not support accounting.

Local accounting does not provide statistics for charging. It only counts and controls the number of concurrent users that use the same local user account. The threshold is configured by using the **access-limit** command.

When the primary accounting method is local, the following rules apply to the accounting of a user:

- The device uses the backup accounting methods in sequence only if local accounting is invalid for one of the following reasons:
  - An exception occurs in the local accounting process.
  - The user account is not configured on the device or the user is not allowed to use the access service.

- The device does not turn to the backup accounting methods if local accounting is invalid because of any other reason. Accounting fails for the user.

Prerequisites

Before configuring accounting methods, complete the following tasks:

1. Determine the access type or service type to be configured. With AAA, you can configure an accounting method for each access type and service type.
2. Determine whether to configure the default accounting method for all access types or service types. The default accounting method applies to all access users. However, the method has a lower priority than the accounting method that is specified for an access type or service type.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter ISP domain view.

   ```
   domain isp-name
   ```

3. (Optional.) Specify default accounting methods for all types of users.

   ```
   accounting default { hwtacacs-scheme hwtacacs-scheme-name
   [ radius-scheme radius-scheme-name ] [ local ] [ none ] | local
   [ radius-scheme radius-scheme-name | hwtacacs-scheme
   hwtacacs-scheme-name ] * [ none ] | none | radius-scheme
   radius-scheme-name [ hwtacacs-scheme hwtacacs-scheme-name ] [ local ]
   [ none ] }
   ```

   By default, the accounting method is **local**.

4. Specify accounting methods for a user type.
   - Specify accounting methods for ADVPN users.

     ```
     accounting advpn { local [ radius-scheme radius-scheme-name ] [ none ]
     | none | radius-scheme radius-scheme-name [ local ] [ none ] }
     ```

     By default, the default accounting methods are used for ADVPN users.

   - Specify the command accounting method.

     ```
     accounting command hwtacacs-scheme hwtacacs-scheme-name
     ```

     By default, the default accounting methods are used for command accounting.

   - Specify accounting methods for IPoE users.

     ```
     accounting ipoe { broadcast radius-scheme radius-scheme-name1
     radius-scheme radius-scheme-name2 [ local ] [ none ] | local
     [ radius-scheme radius-scheme-name ] [ none ] | none | radius-scheme
     radius-scheme-name [ local ] [ none ] }
     ```

     By default, the default accounting methods are used for IPoE users.

   - Specify accounting methods for login users.

     ```
     accounting login { hwtacacs-scheme hwtacacs-scheme-name
     [ radius-scheme radius-scheme-name ] [ local ] [ none ] | local
     [ radius-scheme radius-scheme-name | hwtacacs-scheme
     hwtacacs-scheme-name ] * [ none ] | none | radius-scheme
     radius-scheme-name [ hwtacacs-scheme hwtacacs-scheme-name ]
     [ local ] [ none ] }
     ```

     By default, the default accounting methods are used for login users.

   - Specify accounting methods for portal users.

     ```
     accounting portal { broadcast radius-scheme radius-scheme-name1
     radius-scheme radius-scheme-name2 [ local ] [ none ] | local
     ```

[**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

By default, the default accounting methods are used for portal users.

- Specify accounting methods for PPP users.

  accounting ppp { **broadcast radius-scheme** *radius-scheme-name1* **radius-scheme** *radius-scheme-name2* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] | **hwtacacs-scheme** *hwtacacs-scheme-name* [**radius-scheme** *radius-scheme-name*] [**local**] [**none**] | **local** [**radius-scheme** *radius-scheme-name* | **hwtacacs-scheme** *hwtacacs-scheme-name*] * [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**hwtacacs-scheme** *hwtacacs-scheme-name*] [**local**] [**none**] }

  By default, the default accounting methods are used for PPP users.

- Specify accounting methods for SSL VPN users.

  accounting sslvpn { **local** [**radius-scheme** *radius-scheme-name*] [**none**] | **none** | **radius-scheme** *radius-scheme-name* [**local**] [**none**] }

  By default, the default accounting methods are used for SSL VPN users.

5. (Optional.) Configure extended accounting policies.
   - Configure access control for users that encounter accounting-start failures.

     accounting start-fail { **offline** | **online** }

     By default, the device allows users that encounter accounting-start failures to stay online.

   - Configure access control for users that have failed all their accounting-update attempts.

     accounting update-fail { [**max-times** *max-times*] **offline** | **online** }

     By default, the device allows users that have failed all their accounting-update attempts to stay online.

   - Configure access control for users that have used up their data or time accounting quotas.

     accounting quota-out { **offline** | **online** }

     By default, the device logs off users that have used up their accounting quotas.

Display and maintenance commands for ISP domains

Execute **display** commands in any view.

Task	Command
Display configuration information about an ISP domain or all ISP domains.	**display domain** [*isp-name*]

Configuring domain name delimiters

About this task

A domain name delimiter separates the username part from the domain name part in a username. For the device to correctly extract the username and domain name parts in usernames, you can configure domain name delimiters. Table 3 shows the way that the device interprets a username based on different domain name delimiters.

Table 3 Domain name delimiters and username formats

Domain name delimiter	Username format
At sign (@)	*username@domain-name*
Backslash (\)	*domain-name\username*
Slash (/)	*username/domain-name*
Dot (.)	*username.domain-name*

If a username includes multiple domain name delimiters, the device selects the first delimiter in the search direction specified by using the **domain-delimiter search-from** command.

Restrictions and guidelines

The access module-specific domain name delimiters have higher priority than global domain name delimiters.

Modification of global domain name delimiters or search direction takes effect only on users that come online after the modification.

Procedure

1. Enter system view.

   system-view

2. Configure global domain name delimiters.

   domain-delimiter [**advpn** | **ike** | **ipoe** | **lanaccess** | **login** | **portal** | **ppp** | **sslvpn** | **super**] *string*

   By default, global domain name delimiters include at sign (@), slash (/), and backslash (\).

3. Specify the search direction for the domain name delimiter.

   domain-delimiter search-direction { **backward** | **forward** }

   By default, the search direction is right-to-left.

Setting the maximum number of concurrent login users

About this task

Perform this task to set the maximum number of concurrent users that can log on to the device through a specific protocol, regardless of their authentication methods. The authentication methods include no authentication, local authentication, and remote authentication.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of concurrent login users.

   aaa session-limit { **ftp** | **http** | **https** | **ssh** | **telnet** } *max-sessions*

   By default, the maximum number of concurrent login users is 32 for each user type.

Configuring a NAS-ID

About this task

During RADIUS authentication, the device uses a NAS-ID to set the NAS-Identifier attribute of RADIUS packets so that the RADIUS server can identify the access location of users.

The device selects the NAS-ID for the NAS-Identifier attribute in the following order:

3. NAS-ID bound with VLANs in a NAS-ID profile.
4. NAS-ID in an ISP domain.

If no NAS-ID is configured, the device uses the device name (set by using the **`sysname`** command) as the NAS-ID.

Configuring a NAS-ID profile

About this task

Configure a NAS-ID profile to maintain NAS-ID and VLAN bindings on the device so that the device can send different NAS-Identifier attribute strings in RADIUS requests from different VLANs.

Restrictions and guidelines

You can apply a NAS-ID profile to portal-enabled interfaces. For more information, see "Configuring portal authentication."

You can configure multiple NAS-ID and VLAN bindings in a NAS-ID profile.

A NAS-ID can be bound with more than one VLAN, but a VLAN can be bound with only one NAS-ID. If you configure multiple bindings for the same VLAN, the most recent configuration takes effect.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Create a NAS-ID profile and enter NAS-ID profile view.
   ```
   aaa nas-id profile profile-name
   ```
3. Configure a NAS-ID and VLAN binding in the profile.
   ```
   nas-id nas-identifier bind vlan vlan-id
   ```

Enabling password change prompt logging

About this task

Use this feature to enhance the protection of passwords for Telnet, SSH, HTTP, HTTPS, NETCONF over SSH, and NETCONF over SOAP users and improve the system security.

This feature enables the device to generate logs to prompt users to change their weak passwords at an interval of 24 hours and at the users' login.

A password is a weak password if it does not meet the following requirements:

- Password composition restriction configured by using the **`password-control composition`** command.
- Minimum password length restriction set by using the **`password-control length`** command.
- Password complexity checking policy configured by using the **`password-control complexity`** command.

For a NETCONF over SSH or NETCONF over SOAP user, the device also generates a password change prompt log if any of the following conditions exists:

- The current password of the user is the default password or has expired.
- The user logs in to the device for the first time or uses a new password to log in after global password control is enabled.

The device will no longer generate password change prompt logs for a user when one of the following conditions exists:

- The password change prompt logging feature is disabled.
- The user has changed the password and the new password meets the password control requirements.
- The enabling status of a related password control feature has changed so the current password of the user meets the password control requirements.
- The password composition policy or the minimum password length has changed.

Restrictions and guidelines

You can use the **`display password-control`** command to display password control configuration. For more information about password control commands, see password control commands in *Security Command Reference*.

Procedure

1. Enter system view.

   `system-view`

2. Enable password change prompt logging.

   `local-server log change-password-prompt`

   By default, password change prompt logging is enabled.

Setting the NAS-ID in an ISP domain

1. Enter system view.

   `system-view`

2. Enter ISP domain view.

   `domain` *`isp-name`*

3. Set the NAS-ID in the ISP domain.

   `nas-id` *`nas-identifier`*

   By default, no NAS-ID is set in an ISP domain.

Configuring the device ID

About this task

RADIUS uses the value of the Acct-Session-ID attribute as the accounting ID for a user. The device generates an Acct-Session-ID value that includes the device ID for each online user.

Procedure

1. Enter system view.

   `system-view`

2. Configure the device ID.

   `aaa device-id` *`device-id`*

   By default, the device ID is 0.

Configuring the AAA test feature

About this task

This feature enables the device to send authentication or accounting requests to the specified AAA servers to simulate an authentication or accounting process of a user. Use this feature to identify the reasons for the failure of the interaction between the device and the AAA servers. This feature is applicable only to RADIUS.

When performing an AAA test, the device ignores the status of the specified AAA servers and the RADIUS server load sharing feature. The process of an AAA test is as follows:

1. The device sends authentication requests that carry the specified username and password to the specified authentication server or to the authentication servers in the specified RADIUS scheme. The device tries to communicate with the authentication servers in the specified scheme in sequence.

   The process goes to the next step in the following situations:

   - The device receives an authentication response (no matter the authentication succeeds or fails).
   - The device does not receive any authentication response after making all authentication request attempts.

   This step is skipped if no correct authentication server is specified for the AAA test or no authentication servers are configured in the specified RADIUS scheme.

2. The device sends start-accounting requests to the specified accounting server or to the accounting servers in the specified RADIUS scheme. The device tries to communicate with the accounting servers in the specified scheme in sequence.

   The process goes to the next step in the following situations:

   - The device receives a start-accounting response (no matter the accounting succeeds or fails).
   - The device does not receive any start-accounting response after making all start-accounting request attempts.

   This step and the next step are skipped if no correct accounting server is specified for the AAA test or no accounting servers are configured in the specified RADIUS scheme.

3. The device sends stop-accounting requests to the accounting servers to which it has sent a start-accounting request.

   The process finishes in the following situations:

   - The device receives a stop-accounting response.
   - The device does not receive any stop-accounting response after making all stop-accounting request attempts.

To identify attributes that cause authentication or accounting failures, you can configure the device to carry specific attributes in RADIUS requests or define values for specific attributes in the requests. Table 4 shows the attributes that RADIUS requests carry by default.

Table 4 Attributes that RADIUS requests carry by default

Packet type	Attributes that the type of packets carry by default
RADIUS authentication request	User-Name CHAP-Password (or User-Password) CHAP-Challenge NAS-IP-Address (or NAS-IPv6-Address) Service-Type Framed-Protocol

Packet type	Attributes that the type of packets carry by default
	NAS-Identifier NAS-Port-Type Acct-Session-Id
RADIUS accounting request	User-Name Acct-Status-Type NAS-IP-Address (or NAS-IPv6-Address) NAS-Identifier Acct-Session-Id Acct-Delay-Time Acct-Terminate-Cause

Restrictions and guidelines

When you perform an AAA test, follow these restrictions and guidelines:

- The device might communicate with the AAA servers incorrectly during an AAA test. Make sure no users come online or go offline during an AAA text.
- If the configuration of the specified RADIUS scheme changes, the new configuration does not affect the current AAA test. The modification will take effect in the next test.
- The system can have only one AAA test at a time. Another AAA test can be performed only after the current test finishes.

When you configure attributes to be included in or excluded from RADIUS requests, follow these restrictions and guidelines:

- Before you include an attribute that is already configured to be excluded from RADIUS requests, you must cancel the exclusion configuration by using the **undo exclude** command.
- Before you exclude an attribute that is already configured to be included in RADIUS requests, you must cancel the inclusion configuration by using the **undo include** command.

Prerequisites

Before you perform an AAA test, you must configure a RADIUS scheme that contains the RADIUS servers to be tested.

Plan the RADIUS attributes to be included in RADIUS requests. Besides the attributes carried by default, the device adds the specified attributes to RADIUS packets in the order that they are specified by using the **include** command. Additional attributes cannot be added to a RADIUS request if the length of the RADIUS request will reach or exceed 4096 bytes.

Procedure

1. (Optional.) Configure a RADIUS attribute test group:
   a. Enter system view.
      system-view
   b. Create a RADIUS attribute test group and enter its view.
      radius attribute-test-group *attr-test-group-name*
      You can create multiple RADIUS attribute test groups.
   c. Include an attribute in RADIUS requests.
      include { **accounting** | **authentication** } { **name** *attribute-name* | [**vendor** *vendor-id*] **code** *attribute-code* } **type** { **binary** | **date** | **integer** | **interface-id** | **ip** | **ipv6** | **ipv6-prefix** | **octets** | **string** } value *attribute-value*

For an attribute that RADIUS requests carry by default, use this command to change its attribute value.

d. Exclude an attribute from RADIUS requests.

```
exclude { accounting | authentication } name attribute-name
```

e. Return to system view.

```
quit
```

f. Return to user view.

```
quit
```

2. Perform an AAA test in user view.

```
test-aaa user user-name password password radius-scheme
radius-scheme-name [ radius-server { ipv4-address | ipv6
ipv6-address } port-number [ vpn-instance vpn-instance-name ] ] [ chap
| pap ] [ attribute-test-group attr-test-group-name ] [ trace ]
```

AAA configuration examples

Example: Configuring authentication and authorization for SSH users by a RADIUS server

Network configuration

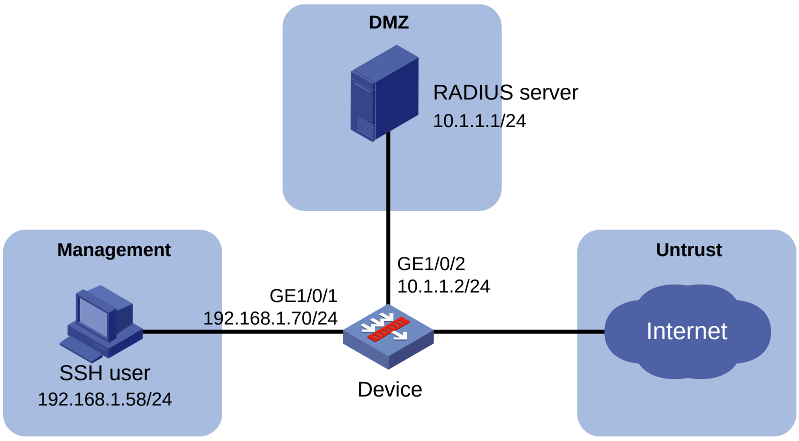
As shown in Figure 11, configure the device to meet the following requirements:

- Use the RADIUS server for SSH user authentication and authorization.
- Include domain names in the usernames sent to the RADIUS server.
- Assign the **network-admin** user role to SSH users after they pass authentication.

The RADIUS server runs IMC PLAT 7.3 (E0605) and IMC UAM 7.3 (E0512). Add an account named **hello@bbb** on the RADIUS server.

The RADIUS server and the device use **expert** as the shared key for secure RADIUS communication. The ports for authentication and accounting are **1812** and **1813**, respectively.

Figure 11 Network diagram



Configuring the RADIUS server

1. Add the device to the IMC Platform as an access device:

Log in to IMC, click the **User** tab, and select **User Access Policy** > **Access Device Management** > **Access Device** from the navigation tree. Then, click **Add** to configure an access device as follows:

a. Set the ports for authentication and accounting to 1812 and 1813, respectively.

b. Select **Device Management Service** from the **Service Type** list.

c. Select **(General)** from the **Access Device Type** list.

d. Set the shared key to **expert** for secure RADIUS communication.

e. Select an access device from the device list or manually add an access device. In this example, the device IP address is 10.1.1.2.

f. Use the default values for other parameters.

g. Click **OK**.

The IP address of the access device specified here must be the same as the source IP address of the RADIUS packets sent from the device. The source IP address is chosen in the following order on the device:

- IP address specified by using the **`nas-ip`** command.
- IP address specified by using the **`radius nas-ip`** command.
- IP address of the outbound interface (the default).

2. Add an account for device management:

Click the **User** tab, and select **Device User** > **Device User** from the navigation tree. Then, click **Add** to configure a device management account as follows:

a. Enter account name **hello@bbb** and set the password.

b. Select **SSH** from the **Login Type** list.

c. Enter **network-admin** in the **Role Name** field.

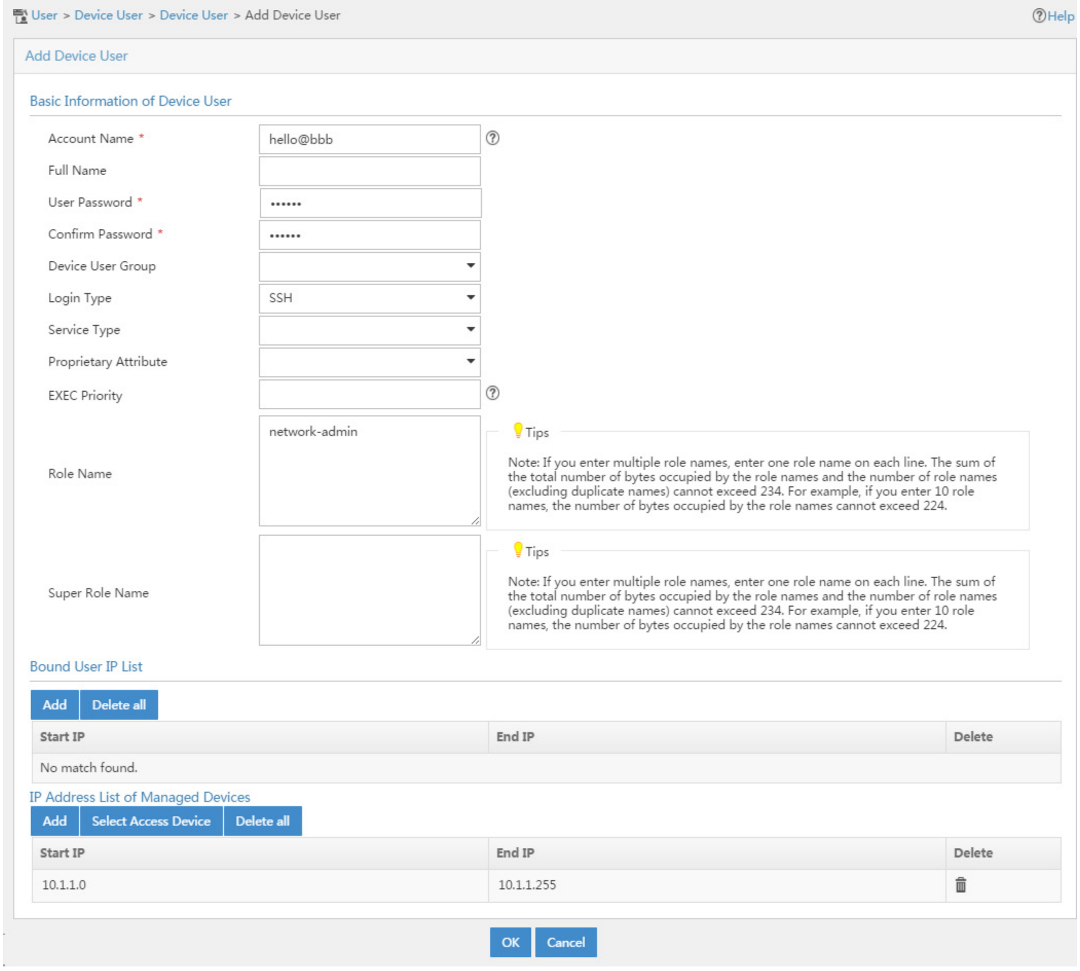
d. Specify 10.1.1.0 to 10.1.1.255 as the IP address range of hosts to be managed.

e. Click **OK**.

NOTE:

The IP address range must contain the IP address of the device.

Figure 12 Adding an account for device management



User > Device User > Device User > Add Device User

Add Device User

Basic Information of Device User

Account Name *	hello@bbb
Full Name	
User Password *	••••••
Confirm Password *	••••••
Device User Group	
Login Type	SSH
Service Type	
Proprietary Attribute	
EXEC Priority	
Role Name	network-admin
Super Role Name	

Tips

Note: If you enter multiple role names, enter one role name on each line. The sum of the total number of bytes occupied by the role names and the number of role names (excluding duplicate names) cannot exceed 234. For example, if you enter 10 role names, the number of bytes occupied by the role names cannot exceed 224.

Tips

Note: If you enter multiple role names, enter one role name on each line. The sum of the total number of bytes occupied by the role names and the number of role names (excluding duplicate names) cannot exceed 234. For example, if you enter 10 role names, the number of bytes occupied by the role names cannot exceed 224.

Bound User IP List

Start IP	End IP	Delete
No match found.		

IP Address List of Managed Devices

Start IP	End IP	Delete
10.1.1.0	10.1.1.255	🗑

Configuring the device

1. Assign IP addresses to interfaces:

   # Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   In this example, the device is directly connected to the SSH user and the RADIUS server.

3. Add interfaces to security zones.

   ```
   [Device] security-zone name Management
   [Device-security-zone-Management] import interface gigabitethernet 1/0/1
   [Device-security-zone-Management] quit
   [Device] security-zone name DMZ
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
   [Device-security-zone-DMZ] quit
   ```

4. Configure a security policy to permit the request packets from the device to the RADIUS server.

   ```
   [Device] security-policy ip
   ```

```
[Device-security-policy-ip] rule name aaalocalout
[Device-security-policy-ip-1-aaalocalout] source-zone local
[Device-security-policy-ip-1-aaalocalout] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-aaalocalout] destination-zone DMZ
[Device-security-policy-ip-1-aaalocalout] destination-ip-host 10.1.1.1
[Device-security-policy-ip-1-aaalocalout] action pass
[Device-security-policy-ip-1-aaalocalout] quit
[Device-security-policy-ip] quit
```

5. Configure key pairs:

   # Create local RSA key pairs, a local DSA key pair, and a local ECDSA key pair.

```
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..
Create the key pair successfully.
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

6. Configure SSH:

   # Enable the Stelnet server, and enable scheme authentication for user lines VTY 0 through VTY 63.

```
[Device] ssh server enable
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
[Device-line-vty0-63] quit
```

7. Configure a RADIUS scheme:

   # Create a RADIUS scheme, configure the authentication server parameters, and configure the device to include domain names in the usernames sent to the RADIUS server.

```
[Device] radius scheme rad
[Device-radius-rad] primary authentication 10.1.1.1 1812
[Device-radius-rad] key authentication simple expert
[Device-radius-rad] user-name-format with-domain
[Device-radius-rad] quit
```

8. Create an ISP domain named **bbb** and configure authentication, authorization, and accounting methods for login users. Because RADIUS user authorization information is piggybacked in authentication responses, the authentication and authorization methods must use the same RADIUS scheme.

```
[Device] domain bbb
[Device-isp-bbb] authentication login radius-scheme rad
[Device-isp-bbb] authorization login radius-scheme rad
[Device-isp-bbb] accounting login none
[Device-isp-bbb] quit
```

# Specify ISP domain **bbb** as the default ISP domain.

```
[Device] domain default enable bbb
```

Verifying the configuration

# Initiate an SSH connection to the device, and enter username **hello@bbb** and the correct password. The user logs in to the device. (Details not shown.)

# Verify that the user can use the commands permitted by the **network-admin** user role. (Details not shown.)

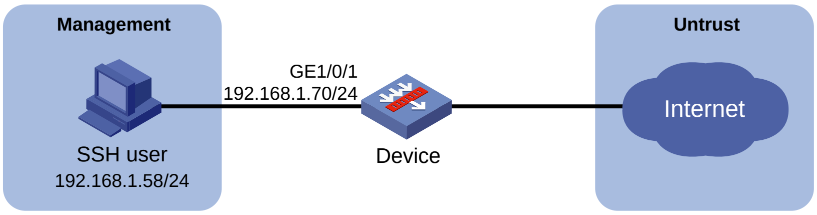
Example: Configuring local authentication and authorization for SSH users

Network configuration

As shown in Figure 13, configure the device to meet the following requirements:

- Perform local authentication and authorization for SSH users.
- Assign the **network-admin** user role to SSH users after they pass authentication.

Figure 13 Network diagram



Procedure

1. Assign an IP address to GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

2. Configure settings for routing.

   In this example, the device is directly connected to the SSH user.

3. Add interface GigabitEthernet 1/0/1 to security zone **Management**.

```
[Device] security-zone name Management
[Device-security-zone-Management] import interface gigabitethernet 1/0/1
[Device-security-zone-Management] quit
```

4. Configure key pairs:

   # Create local RSA key pairs, a local DSA key pair, and a local ECDSA key pair.

```
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
```

```
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..
Create the key pair successfully.
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

5. Configure SSH:

   # Enable the Stelnet server, and enable scheme authentication for user lines VTY 0 through VTY 63.

   ```
   [Device] ssh server enable
   [Device] line vty 0 63
   [Device-line-vty0-63] authentication-mode scheme
   [Device-line-vty0-63] quit
   ```

6. Configure a local SSH user:

   # Create a device management user, and configure the service type, password, and user role for the user.

   ```
   [Device] local-user ssh class manage
   [Device-luser-manage-ssh] service-type ssh
   [Device-luser-manage-ssh] password simple 123456TESTplat&!
   [Device-luser-manage-ssh] authorization-attribute user-role network-admin
   [Device-luser-manage-ssh] quit
   ```

7. Configure an ISP domain:

   # Create an ISP domain named **bbb** and configure the domain to use local authentication and authorization for login users.

   ```
   [Device] domain bbb
   [Device-isp-bbb] authentication login local
   [Device-isp-bbb] authorization login local
   [Device-isp-bbb] quit
   ```

   # Specify ISP domain **bbb** as the default ISP domain.

   ```
   [Device] domain default enable bbb
   ```

Verifying the configuration

# Initiate an SSH connection to the device, and enter username **ssh@bbb** and the correct password. The user logs in to the device. (Details not shown.)

# Verify that the user can use the commands permitted by the network-admin user role. (Details not shown.)

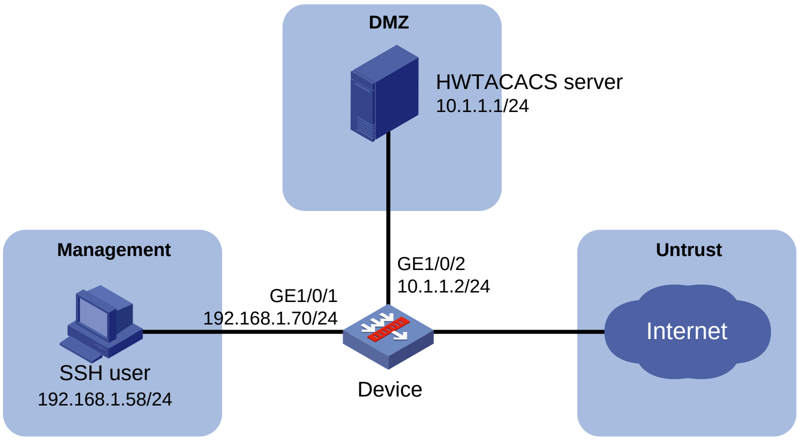
Example: Configuring AAA for SSH users by an HWTACACS server

Network configuration

As shown in Figure 14, configure the device to meet the following requirements:

- Use the HWTACACS server for SSH user authentication, authorization, and accounting.
- Assign the default user role **network-operator** to SSH users after they pass authentication.
- Exclude domain names from the usernames sent to the HWTACACS server.
- Use **expert** as the shared keys for secure HWTACACS communication.

Figure 14 Network diagram



Configuring the HWTACACS server

# Set the shared keys to **expert** for secure communication with the device, add an account for the SSH user, and specify the password. (Details not shown.)

Configuring the device

1. Assign IP addresses to interfaces:

   # Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   In this example, the device is directly connected to the SSH user and the HWTACACS server.

3. Add interfaces to security zones.

   ```
   [Device] security-zone name Management
   [Device-security-zone-Management] import interface gigabitethernet 1/0/1
   [Device-security-zone-Management] quit
   [Device] security-zone name DMZ
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
   [Device-security-zone-DMZ] quit
   ```

4. Configure a security policy to permit the request packets from the device to the HWTACACS server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name aaalocalout
[Device-security-policy-ip-1-aaalocalout] source-zone local
[Device-security-policy-ip-1-aaalocalout] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-aaalocalout] destination-zone DMZ
[Device-security-policy-ip-1-aaalocalout] destination-ip-host 10.1.1.1
[Device-security-policy-ip-1-aaalocalout] action pass
[Device-security-policy-ip-1-aaalocalout] quit
[Device-security-policy-ip] quit
```

5. Configure key pairs:

   # Create local RSA key pairs, a local DSA key pair, and a local ECDSA key pair.

```
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..
Create the key pair successfully.
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

6. Configure SSH:

   # Enable the Stelnet server, and enable scheme authentication for user lines VTY 0 through VTY 63.

```
[Device] ssh server enable
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
[Device-line-vty0-63] quit
```

   # Enable the default user role feature to assign authenticated SSH users the default user role **network-operator**.

```
[Device] role default-role enable
```

7. Configure an HWTACACS scheme:

   # Create an HWTACACS scheme, configure the HWTACACS server parameters, and configure the device to exclude domain names from the usernames sent to the HWTACACS server.

```
[Device] hwtacacs scheme hwtac
[Device-hwtacacs-hwtac] primary authentication 10.1.1.1 49
[Device-hwtacacs-hwtac] primary authorization 10.1.1.1 49
```

```
[Device-hwtacacs-hwtac] primary accounting 10.1.1.1 49
[Device-hwtacacs-hwtac] key authentication simple expert
[Device-hwtacacs-hwtac] key authorization simple expert
[Device-hwtacacs-hwtac] key accounting simple expert
[Device-hwtacacs-hwtac] user-name-format without-domain
[Device-hwtacacs-hwtac] quit
```

8. Configure an ISP domain:

   # Create an ISP domain named **bbb**, and configure the domain to use the HWTACACS scheme for login user authentication, authorization, and accounting.

   ```
   [Device] domain bbb
   [Device-isp-bbb] authentication login hwtacacs-scheme hwtac
   [Device-isp-bbb] authorization login hwtacacs-scheme hwtac
   [Device-isp-bbb] accounting login hwtacacs-scheme hwtac
   [Device-isp-bbb] quit
   ```

   # Specify ISP domain **bbb** as the default ISP domain.

   ```
   [Device] domain default enable bbb
   ```

Verifying the configuration

# Initiate an SSH connection to the device, and enter the correct username and password. The user logs in to the device. (Details not shown.)

# Verify that the user can use the commands permitted by the network-operator user role. (Details not shown.)

Example: Configuring authentication for SSH users by an LDAP server

Network configuration

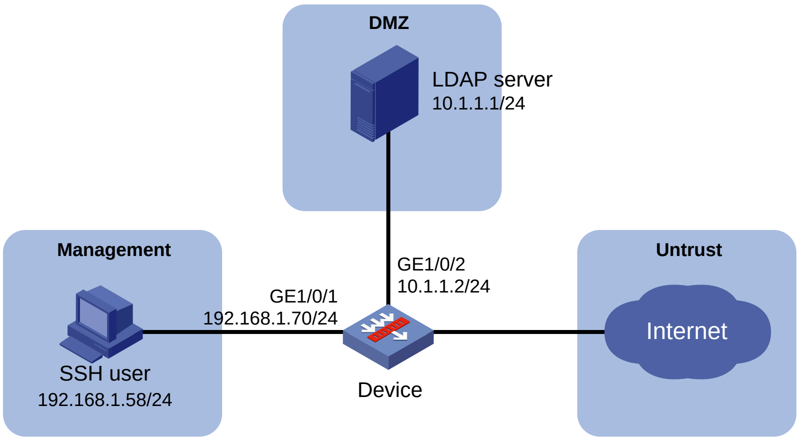
As shown in Figure 15, the LDAP server runs Microsoft Windows 2003 Server Active Directory and uses domain **ldap.com**.

Configure the device to meet the following requirements:

- Use the LDAP server to authenticate SSH users.
- Assign the level-0 user role to SSH users after they pass authentication.

On the LDAP server, set the administrator password to **admin!123456**, add a user named **aaa**, and set the user's password to **ldap!123456**.

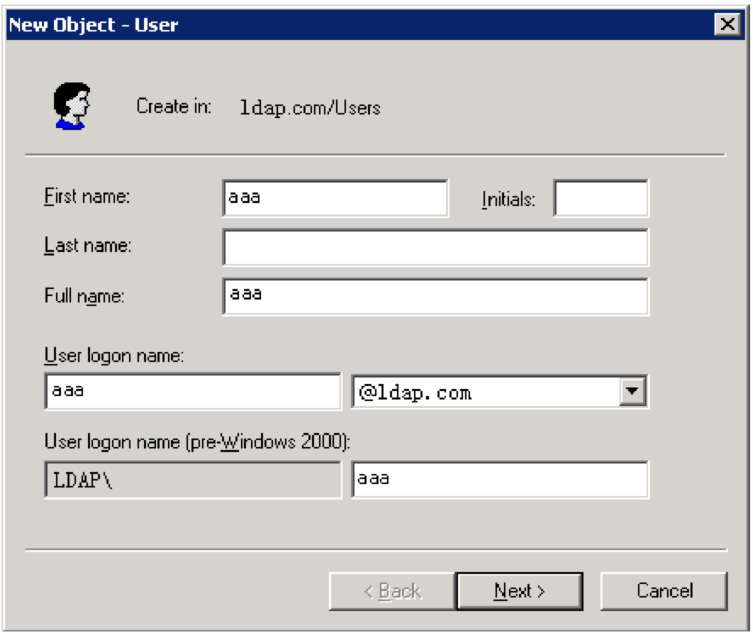
Figure 15 Network diagram



Configuring the LDAP server

1. Add a user named **aaa** and set the password to **ldap!123456**:
   a. On the LDAP server, select **Start** > **Control Panel** > **Administrative Tools**.
   b. Double-click **Active Directory Users and Computers**.
   The **Active Directory Users and Computers** window is displayed.
   c. From the navigation tree, click **Users** under the **ldap.com** node.
   d. Select **Action** > **New** > **User** from the menu to display the dialog box for adding a user.
   e. Enter logon name **aaa** and click **Next**.

Figure 16 Adding user aaa



Create in: ldap.com/Users

First name: aaa

Initials:

Last name:

Full name: aaa

User logon name:

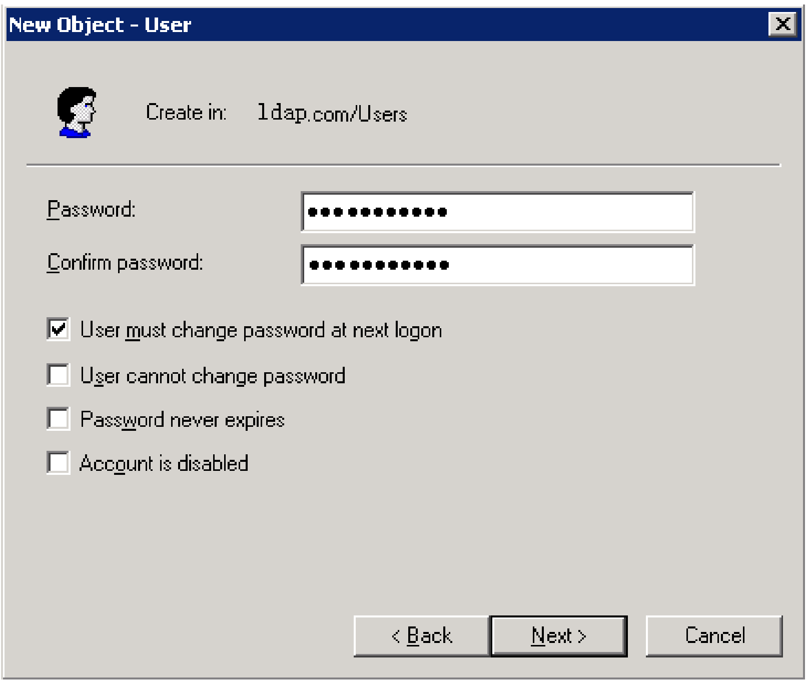
aaa @ldap.com

User logon name (pre-Windows 2000):

LDAP\ aaa

   f. In the dialog box, enter password **ldap!123456**, select options as needed, and click **Next**.

Figure 17 Setting the user's password



New Object - User

Create in: ldap.com/Users

Password: ●●●●●●●●●●●●

Confirm password: ●●●●●●●●●●●●

☑ User must change password at next logon

□ User cannot change password

□ Password never expires

□ Account is disabled

g. Click **OK**.

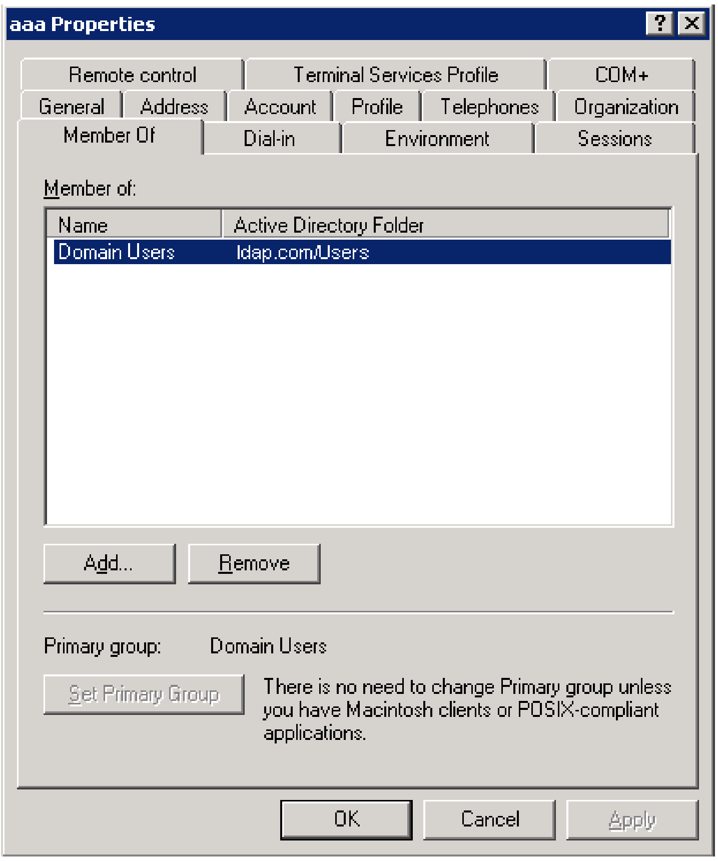
2. Add user **aaa** to group **Users**:

a. From the navigation tree, click **Users** under the **ldap.com** node.

b. In the right pane, right-click user **aaa** and select **Properties**.

c. In the dialog box, click the **Member Of** tab and click **Add**.

Figure 18 Modifying user properties



Member of:

Name	Active Directory Folder
Domain Users	ldap.com/Users

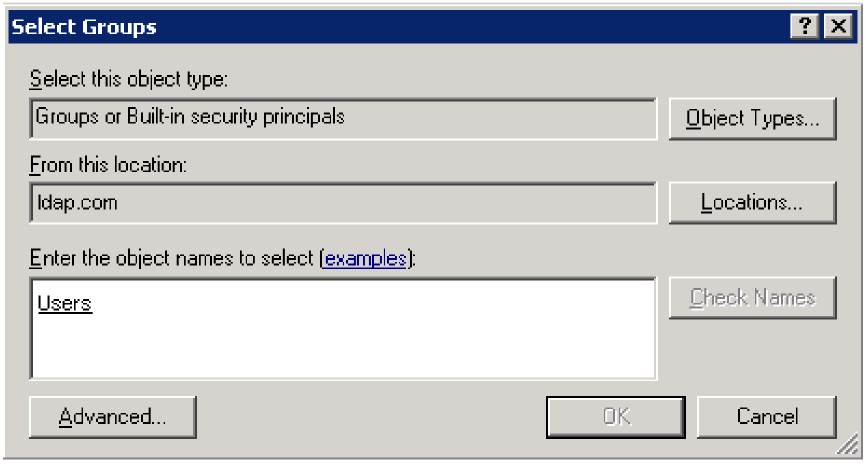
Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

d. In the **Select Groups** dialog box, enter **Users** in the **Enter the object names to select** field, and click **OK**.

User **aaa** is added to group **Users**.

Figure 19 Adding user aaa to group Users



Select this object type:
Groups or Built-in security principals

From this location:
ldap.com

Enter the object names to select (examples):
Users

3. Set the administrator password to **admin!123456**:

   a. In the right pane, right-click user **Administrator** and select **Set Password**.

   b. In the dialog box, enter the administrator password. (Details not shown.)

Configuring the device

1. Assign IP addresses to interfaces:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

In this example, the device is directly connected to the SSH user and the LDAP server.

3. Add interfaces to security zones.

```
[Device] security-zone name Management
[Device-security-zone-Management] import interface gigabitethernet 1/0/1
[Device-security-zone-Management] quit
[Device] security-zone name DMZ
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
```

4. Configure a security policy to permit the request packets from the device to the LDAP server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name laaalocalout
[Device-security-policy-ip-1-aaalocalout] source-zone local
[Device-security-policy-ip-1-aaalocalout] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-aaalocalout] destination-zone DMZ
[Device-security-policy-ip-1-aaalocalout] destination-ip-host 10.1.1.1
[Device-security-policy-ip-1-aaalocalout] action pass
[Device-security-policy-ip-1-aaalocalout] quit
[Device-security-policy-ip] quit
```

5. Configure key pairs:

# Create local RSA key pairs, a local DSA key pair, and a local ECDSA key pair.

```
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..
Create the key pair successfully.
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

6. Configure SSH:

# Enable the Stelnet server, and enable scheme authentication for user lines VTY 0 through VTY 63.

```
[Device] ssh server enable
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
[Device-line-vty0-63] quit
```

7. Configure LDAP:

# Configure an LDAP server, specify the IP address of the LDAP server, the administrator DN, and the administrator password, and configure the base DN for user search.

```
[Device] ldap server ldap1
[Device-ldap-server-ldap1] ip 10.1.1.1
[Device-ldap-server-ldap1] login-dn cn=administrator,cn=users,dc=ldap,dc=com
[Device-ldap-server-ldap1] login-password simple admin!123456
[Device-ldap-server-ldap1] search-base-dn dc=ldap,dc=com
[Device-ldap-server-ldap1] quit
```

# Create an LDAP scheme, and specify the LDAP authentication server.

```
[Device] ldap scheme ldap1-shml
[Device-ldap-ldap-shml] authentication-server ldap1
[Device-ldap-ldap1-shml] quit
```

8. Configure an ISP domain:

# Create an ISP domain named **bbb** and configure the authentication, authorization, and accounting methods for login users.

```
[Device] domain bbb
[Device-isp-bbb] authentication login ldap-scheme ldap1-shml
[Device-isp-bbb] authorization login none
[Device-isp-bbb] accounting login none
[Device-isp-bbb] quit
```

# Specify ISP domain **bbb** as the default ISP domain.

```
[Device] domain default enable bbb
```

Verifying the configuration

# Initiate an SSH connection to the device, and enter username **aaa@bbb** and password **ldap!123456**. The user logs in to the device. (Details not shown.)

# Verify that the user can use the commands permitted by the level-0 user role. (Details not shown.)

Example: Configuring authentication and authorization for SSL VPN users by an LDAP server

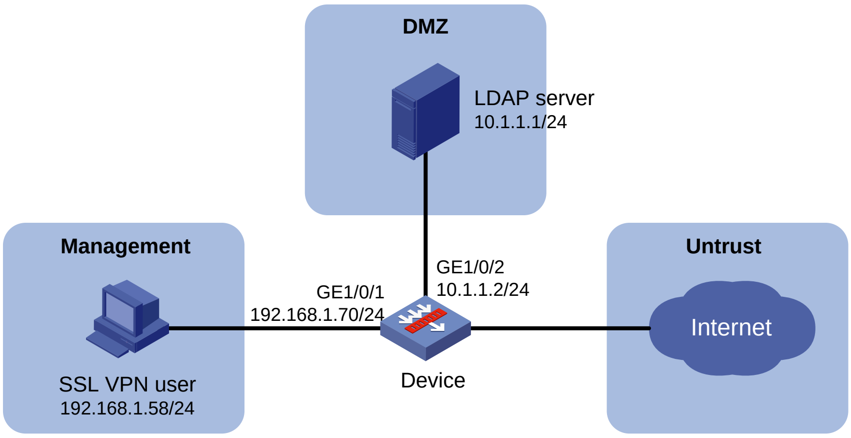
Network configuration

As shown in Figure 20, configure the device to meet the following requirements:

- Use the LDAP server to perform authentication and authorization for the SSL VPN user.
- Act as an SSL VPN gateway. The gateway IP address is 192.168.1.70 and the service port number is 8080.

The LDAP server runs Microsoft Windows 2003 Server Active Directory and uses domain **ldap.com**. The server assigns an SSL VPN policy group named **pg1** to the user after authentication. The policy group specifies the Web resources that the user can access.

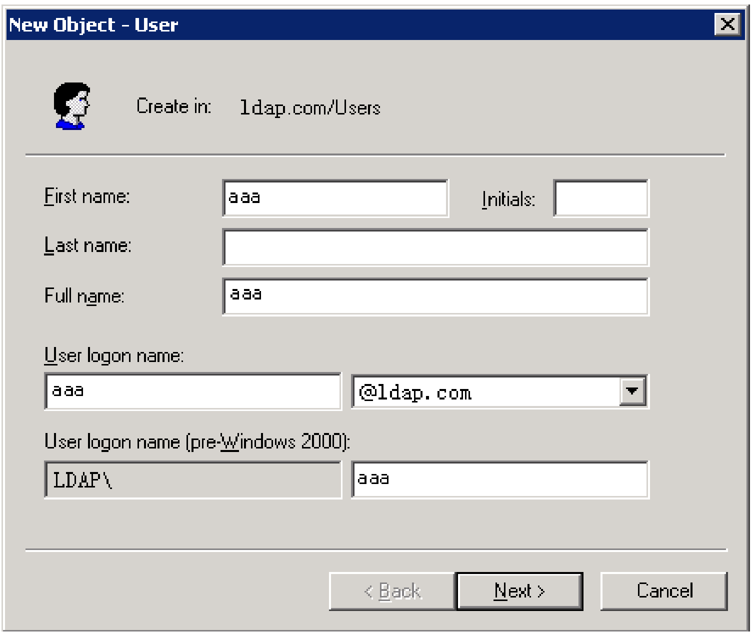
Figure 20 Network diagram



Configuring the LDAP server

1. Add a user named **aaa** and set the password to **ldap!123456**:
   a. On the LDAP server, select **Start** > **Control Panel** > **Administrative Tools**.
   b. Double-click **Active Directory Users and Computers**.
      The **Active Directory Users and Computers** window is displayed.
   c. From the navigation tree, click **Users** under the **ldap.com** node.
   d. Select **Action** > **New** > **User** from the menu to display the dialog box for adding a user.
   e. Enter logon name **aaa** and click **Next**.

      Figure 21 Adding user aaa



New Object - User

Create in: ldap.com/Users

First name: aaa

Initials:

Last name:

Full name: aaa

User logon name:

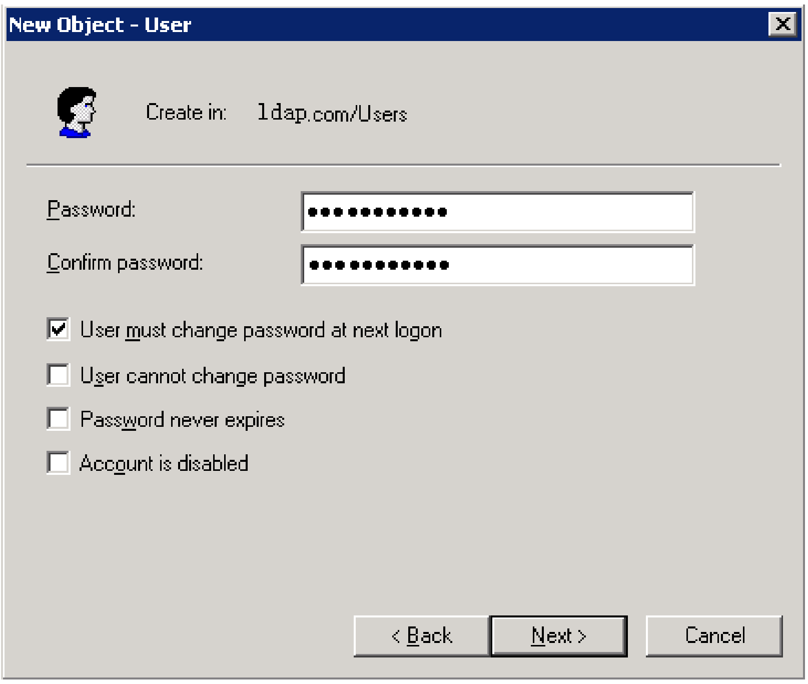
aaa @ldap.com

User logon name (pre-Windows 2000):

LDAP\ aaa

   f. In the dialog box, enter password **ldap!123456**, select options as needed, and click **Next**.

Figure 22 Setting the user's password



New Object - User

Create in: ldap.com/Users

Password: ●●●●●●●●●●●●

Confirm password: ●●●●●●●●●●●●

☑ User must change password at next logon

□ User cannot change password

□ Password never expires

□ Account is disabled

g. Click **OK**.

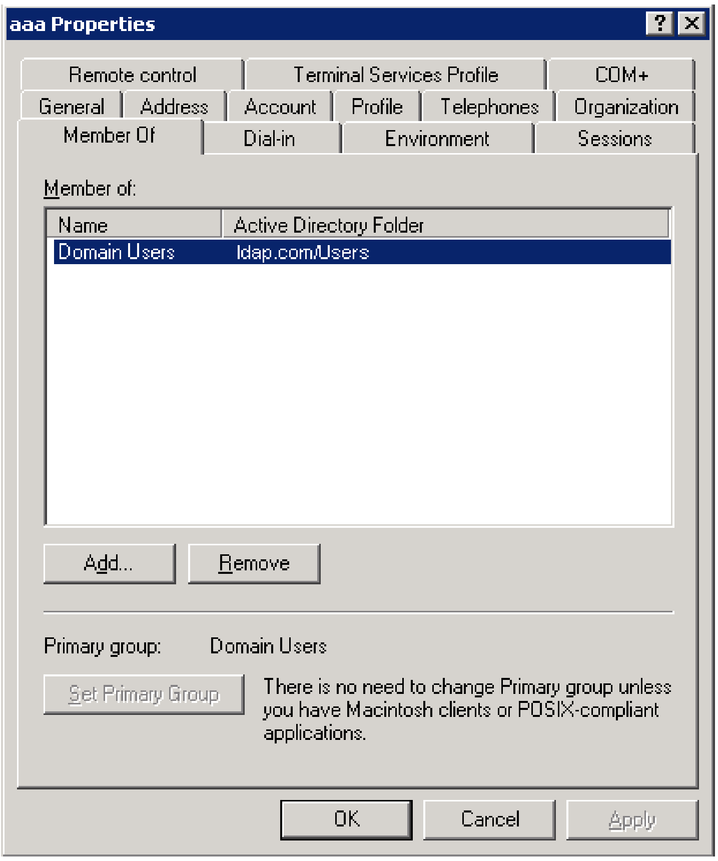
2. Add user **aaa** to group **Users**:

a. From the navigation tree, click **Users** under the **ldap.com** node.

b. In the right pane, right-click user **aaa** and select **Properties**.

c. In the dialog box, click the **Member Of** tab and click **Add**.

Figure 23 Modifying user properties



Member of:

Name	Active Directory Folder
Domain Users	ldap.com/Users

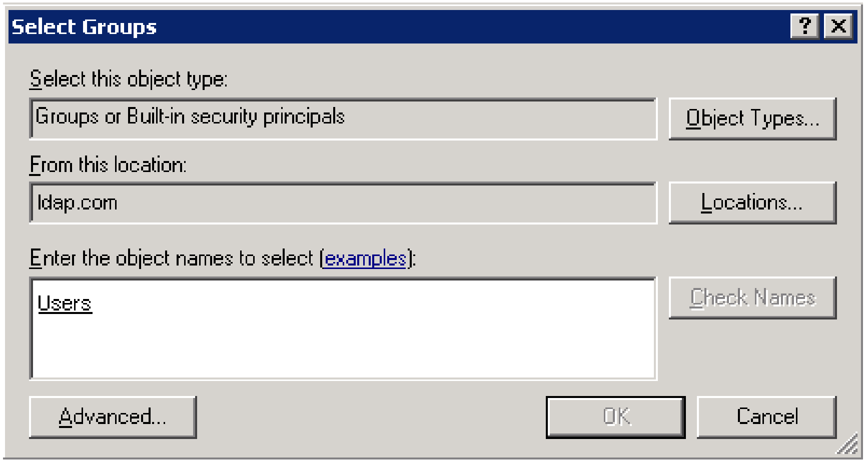
Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

d. In the **Select Groups** dialog box, enter **Users** in the **Enter the object names to select** field, and click **OK**.

User **aaa** is added to group **Users**.

Figure 24 Adding user aaa to group Users



Select this object type:
Groups or Built-in security principals

From this location:
ldap.com

Enter the object names to select (examples):
Users

3. Set the administrator password to **admin!123456**:

   a. In the right pane, right-click user **Administrator** and select **Set Password**.

   b. In the dialog box, enter the administrator password. (Details not shown.)

Configuring the device

1. Assign IP addresses to interfaces:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   In this example, the device is directly connected to the SSL VPN user and the LDAP server.

3. Add interfaces to security zones.

```
[Device] security-zone name Trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name DMZ
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
[Device] security-zone name Untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

   # Configure a rule to permit the traffic from the SSL VPN user to the device.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name sslvpnlocalin
[Device-security-policy-ip-1-sslvpnlocalin] source-zone Trust
[Device-security-policy-ip-1-sslvpnlocalin] destination-zone Local
[Device-security-policy-ip-1-sslvpnlocalin] action pass
[Device-security-policy-ip-1-sslvpnlocalin] quit
```

# Configure a rule to permit the traffic from the device to the SSL VPN user.

```
[Device-security-policy-ip] rule name sslvpnlocalout
[Device-security-policy-ip-2-sslvpnlocalout] source-zone Local
[Device-security-policy-ip-2-sslvpnlocalout] destination-zone Trust
[Device-security-policy-ip-2-sslvpnlocalout] action pass
[Device-security-policy-ip-2-sslvpnlocalout] quit
```

# Configure a rule to permit the traffic from the device to the LDAP server.

```
[Device-security-policy-ip] rule name ldaplocalout
[Device-security-policy-ip-3-ldaplocalout] source-zone local
[Device-security-policy-ip-3-ldaplocalout] source-ip-host 10.1.1.2
[Device-security-policy-ip-3-ldaplocalout] destination-zone DMZ
[Device-security-policy-ip-3-ldaplocalout] destination-ip-host 10.1.1.1
[Device-security-policy-ip-3-ldaplocalout] action pass
[Device-security-policy-ip-3-ldaplocalout] quit
```

# Configure a rule to permit the traffic from the device to the **Untrust** security zone.

```
[Device-security-policy-ip] rule name accesslocalout
[Device-security-policy-ip-4-accesslocalout] source-zone Local
[Device-security-policy-ip-4-accesslocalout]destination-zone Untrust
[Device-security-policy-ip-4-accesslocalout] action pass
[Device-security-policy-ip-4-accesslocalout] quit
```

# Configure a rule to permit the traffic from the **Untrust** security zone to the device.

```
[Device-security-policy-ip] rule name returnlocalin
[Device-security-policy-ip-5-returnlocalin] source-zone Untrust
[Device-security-policy-ip-5-returnlocalin] destination-zone Local
[Device-security-policy-ip-5-returnlocalin] action pass
[Device-security-policy-ip-5-returnlocalin] quit
[Device-security-policy-ip] quit
```

5. Create a PKI domain named **sslvpn** and obtain the CA and local certificates (see "Configuring PKI"). (Details not shown.)
6. Configure SSL:

   # Create an SSL server policy named **myssl**, and specify PKI domain **sslvpn** for the SSL server policy.

```
[Device] ssl server-policy myssl
[Device-server-policy-myssl] pki-domain sslvpn
[Device-server-policy-myssl] quit
```

7. Configure SSL VPN:

   # Create and configure SSL VPN gateway name **g1**.

```
[Device] sslvpn gateway g1
[Device-sslvpn-gateway-g1] ssl server-policy myssl
[Device-sslvpn-gateway-g1] ip address 192.168.1.70 port 8080
[Device-sslvpn-gateway-g1] service enable
[Device-sslvpn-gateway-g1] quit
```

   # Create an SSL VPN context named **aaa**, specify gateway **g1** for the SSL VPN context, and specify domain **bbb** for authentication, authorization, and accounting of SSL VPN users in the context.

```
[Device] sslvpn context aaa
[Device-sslvpn-context-aaa] gateway g1
[Device-sslvpn-context-aaa] aaa domain bbb
```

   # Create a URL item named **urlitem** and specify the resource URL in the URL item.

```
[Device-sslvpn-context-aaa] url-item urlitem
[Device-sslvpn-context-aaa-url-item-urlitem] url http://20.2.2.2
[Device-sslvpn-context-aaa-url-item-urlitem] quit
[Device-sslvpn-context-aaa] url-list urllist
[Device-sslvpn-context-aaa-url-list-urllist] heading web
[Device-sslvpn-context-aaa-url-list-urllist] resources url-item urlitem
[Device-sslvpn-context-aaa-url-list-urllist] quit
```

   # Create an SSL VPN policy group named **pg1** for SSL VPN context **aaa**, add URL list **urllist** to the policy group for Web access, and enable the SSL VPN context.

```
[Device-sslvpn-context-aaa] policy-group pg1
[Device-sslvpn-context-aaa-policy-group-pg1] resources url-list urllist
[Device-sslvpn-context-aaa-policy-group-pg1] quit
[Device-sslvpn-context-aaa] service enable
[Device-sslvpn-context-aaa] quit
```

8. Configure LDAP:

   # Configure LDAP server **ldap1**, specify the IP address of the server, the administrator DN, and the administrator password, and configure the base DN for user search.

```
[Device] ldap server ldap1
[Device-ldap-server-ldap1] ip 10.1.1.1
[Device-ldap-server-ldap1] login-dn cn=administrator,cn=users,dc=ldap,dc=com
```

```
[Device-ldap-server-ldap1] login-password simple admin!123456
[Device-ldap-server-ldap1] search-base-dn dc=ldap,dc=com
[Device-ldap-server-ldap1] quit
```

# Create an LDAP attribute map named **test.**, and map a partial value string of the LDAP attribute named **memberof** to AAA attribute named **user-group**.

```
[Device] ldap attribute-map test
[Device-ldap-attr-map-test] map ldap-attribute memberof prefix cn= delimiter,
aaa-attribute user-group
[Device-ldap-attr-map-test] quit
```

# Create an LDAP scheme, specify the LDAP authentication and authorization servers, and specify LDAP attribute map **test** in the LDAP scheme.

```
[Device] ldap scheme shml
[Device-ldap-shml] authentication-server ldap1
[Device-ldap-shml] authorization-server ldap1
[Device-ldap-shml] attribute-map test
[Device-ldap-shml] quit
```

9. Create an ISP domain named **bbb** and configure the authentication, authorization, and accounting methods for SSL VPN users.

```
[Device] domain bbb
[Device-isp-bbb] authentication sslvpn ldap-scheme shml
[Device-isp-bbb] authorization sslvpn ldap-scheme shml
[Device-isp-bbb] accounting sslvpn none
[Device-isp-bbb] quit
```

10. Create a user group named **users** and authorize SSL VPN policy group **pg1** to the group.

```
[Device] user-group users
[Device-ugroup-users] authorization-attribute sslvpn-policy-group pg1
[Device-ugroup-users] quit
```

Verifying the configuration

# In the Web browser, enter **https://192.168.1.70:8080** in the address bar.

# Enter username **aaa@bbb** and password **ldap!123456**. The user logs in to the website. (Details not shown.)

# Verify that the user can access the Web resources in SSL VPN policy group **pg1**.

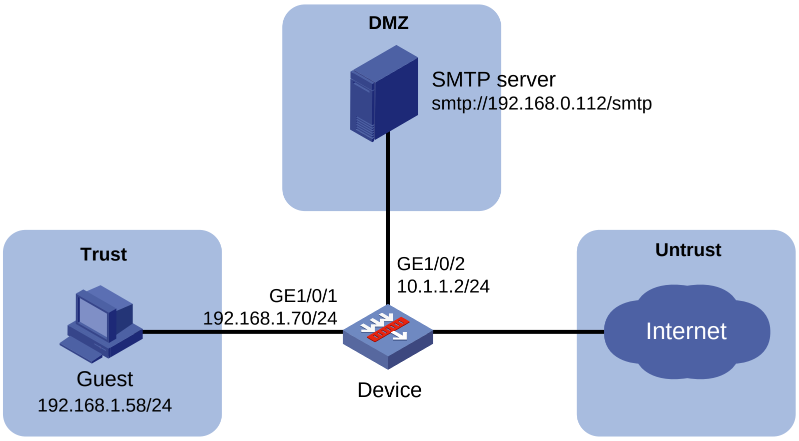
Example: Configuring and managing a local guest

Network configuration

As shown in Figure 25, create a local guest named **user1** for Jack. Configure local guest attributes and manage the local guest on the device as follows:

- Configure attributes for the local guest, including the password, user group, validity period, and sponsor information.
- Enable the guest auto-delete feature.
- Specify an SMTP server and email sender address for the device to send local guest email notifications.
- Configure email addresses for the local guest, guest sponsor, and guest manager.
- Configure the subject and body of the email notifications to be sent to the guest, guest sponsor, and guest manager.
- Send email notifications of the local guest account information to the guest and guest sponsor.

Figure 25 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.70 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   In this example, the device is directly connected to the guest and the SMTP server.

3. Add interfaces to security zones.

   ```
   [Device] security-zone name Trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name DMZ
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
   [Device-security-zone-DMZ] quit
   ```

4. Configure a security policy:

   # Configure a rule named **guestlocalin** to permit the traffic from the guest to the device.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name guestlocalin
   [Device-security-policy-ip-1-guestlocalin] source-zone Trust
   [Device-security-policy-ip-1-guestlocalin] destination-zone Local
   [Device-security-policy-ip-1-guestlocalin] action pass
   [Device-security-policy-ip-1-guestlocalin] quit
   ```

   # Configure a rule named **guestlocalout** to permit the traffic from the device to the guest.

   ```
   [Device-security-policy-ip] rule name guestlocalout
   [Device-security-policy-ip-2-guestlocalout] source-zone Local
   [Device-security-policy-ip-2-guestlocalout] destination-zone Trust
   [Device-security-policy-ip-2-guestlocalout] action pass
   [Device-security-policy-ip-2-guestlocalout] quit
   ```

   # Configure a rule named **smtplocalout**to permit the traffic from the device to the SMTP server.

   ```
   [Device-security-policy-ip] rule name smtplocalout
   ```

```
[Device-security-policy-ip-3-smtplocalout] source-zone local
[Device-security-policy-ip-3-smtplocalout] destination-zone DMZ
[Device-security-policy-ip-3-smtplocalout] action pass
[Device-security-policy-ip-3-smtplocalout] quit
```

# Configure a rule named **smtplocalin** to permit the traffic from the SMTP server to the device.

```
[Device-security-policy-ip] rule name smtplocalin
[Device-security-policy-ip-4-smtplocalin] source-zone DMZ
[Device-security-policy-ip-4-smtplocalin] destination-zone Local
[Device-security-policy-ip-4-smtplocalin] action pass
[Device-security-policy-ip-4-smtplocalin] quit
[Device-security-policy-ip] quit
```

5. Manage local guests:

# Enable the guest auto-delete feature for expired local guests.

```
[Device] local-guest auto-delete enable
```

# Specify an SMTP server to send local guest email notifications.

```
[Device] local-guest email smtp-server smtp://192.168.0.112/smtp
```

# Specify the email sender address as **bbb@ccc.com** in the email notifications sent by the device for local guests.

```
[Device] local-guest email sender bbb@ccc.com
```

# Specify the email address of the guest manager as **guest-manager@ccc.com**.

```
[Device] local-guest manager-email guest-manager@ccc.com
```

# Configure the subject and body of the email notifications to be sent to the local guest.

```
[Device] local-guest email format to guest subject Guest account information
[Device] local-guest email format to guest body A guest account has been created for
your use. The username, password, and valid dates for the account are given below.
```

# Configure the subject and body of the email notifications to be sent to the guest sponsor.

```
[Device] local-guest email format to sponsor subject Guest account information
[Device] local-guest email format to sponsor body A guest account has been created.
The username, password, and valid dates for the account are given below.
```

# Configure the subject and body of the email notifications to be sent to the guest manager.

```
[Device] local-guest email format to manager subject Guest registration information
[Device] local-guest email format to manager body A guest account has been registered.
The username for the account is given below. Please approve the register information.
```

6. Configure the local guest:

# Create a user group named **guest1**.

```
[Device] user-group guest1
[Device-ugroup-guest1] quit
```

# Create a local guest named **user1**, configure the password of the guest, and configure the user group, name, company, email address, phone number, description, and validity period.

```
[Device] local-user user1 class network guest
[Device-luser-network(guest)-user1] password simple 123456
[Device-luser-network(guest)-user1] group guest1
[Device-luser-network(guest)-user1] full-name Jack
[Device-luser-network(guest)-user1] company cc
[Device-luser-network(guest)-user1] email Jack@cc.com
[Device-luser-network(guest)-user1] phone 131129237
[Device-luser-network(guest)-user1] description A guest from company cc
```

```
[Device-luser-network(guest)-user1] validity-datetime from 2015/4/1 08:00:00 to
2015/4/3 18:00:00
```

# Configure the name, email address, and department of the guest sponsor.

```
[Device-luser-network(guest)-user1] sponsor-full-name Sam
[Device-luser-network(guest)-user1] sponsor-email Sam@aa.com
[Device-luser-network(guest)-user1] sponsor-department security
[Device-luser-network(guest)-user1] quit
[Device] quit
```

7. Configure the device to send guest email notifications:

   # Send an email notification to the guest sponsor.

```
<Device> local-guest send-email user-name user1 to sponsor
```

   # Send an email notification to the guest.

```
<Device> local-guest send-email user-name user1 to guest
```

Verifying the configuration

# Display local guest information.

```
<Device> display local-user user-name user1 class network guest
Total 1 local users matched.

Network access guest user user1:
  State:                     Active
  Service type:              LAN access/Portal
  User group:                guest1
  Full name:                 Jack
  Company:                   cc
  Email:                     Jack@cc.com
  Phone:                     131129237
  Description:               A guest from company cc
  Sponsor full name:         Sam
  Sponsor department:        security
  Sponsor email:             Sam@aa.com
  Period of validity:
    Start date and time:     2015/04/01-08:00:00
    Expiration date and time:2015/04/03-18:00:00
```

# Verify that Jack can use username **user1** and password **123456** to pass local authentication and come online during the validity period. (Details not shown.)

Troubleshooting AAA

RADIUS authentication failure

Symptom

User authentication always fails.

Analysis

Possible reasons include:

- A communication failure exists between the NAS and the RADIUS server.

- The username is not in the *userid@isp-name* format, or the ISP domain is not correctly configured on the NAS.
- The user is not configured on the RADIUS server.
- The password entered by the user is incorrect.
- The RADIUS server and the NAS are configured with different shared keys.

Solution

To resolve the problem:

1. Verify the following items:
   - The NAS and the RADIUS server can ping each other.
   - The username is in the *userid@isp-name* format and the ISP domain is correctly configured on the NAS.
   - The user is configured on the RADIUS server.
   - The correct password is entered.
   - The same shared key is configured on both the RADIUS server and the NAS.
2. If the problem persists, contact NSFOCUS Support.

RADIUS packet delivery failure

Symptom

RADIUS packets cannot reach the RADIUS server.

Analysis

Possible reasons include:

- A communication failure exists between the NAS and the RADIUS server.
- The NAS is not configured with the IP address of the RADIUS server.
- The authentication and accounting UDP ports configured on the NAS are incorrect.
- The RADIUS server's authentication and accounting port numbers are being used by other applications.

Solution

To resolve the problem:

1. Verify the following items:
   - The link between the NAS and the RADIUS server works well at both the physical and data link layers.
   - The IP address of the RADIUS server is correctly configured on the NAS.
   - The authentication and accounting UDP port numbers configured on the NAS are the same as those of the RADIUS server.
   - The RADIUS server's authentication and accounting port numbers are available.
2. If the problem persists, contact NSFOCUS Support.

RADIUS accounting error

Symptom

A user is authenticated and authorized, but accounting for the user is not normal.

Analysis

The accounting server configuration on the NAS is not correct. Possible reasons include:

- The accounting port number configured on the NAS is incorrect.
- The accounting server IP address configured on the NAS is incorrect. For example, the NAS is configured to use a single server to provide authentication, authorization, and accounting services, but in fact the services are provided by different servers.

Solution

To resolve the problem:

1. Verify the following items:
   - The accounting port number is correctly configured.
   - The accounting server IP address is correctly configured on the NAS.
2. If the problem persists, contact NSFOCUS Support.

Troubleshooting HWTACACS

Similar to RADIUS troubleshooting. See "RADIUS authentication failure", "RADIUS packet delivery failure", and "RADIUS accounting error."

LDAP authentication failure

Symptom

User authentication fails.

Analysis

Possible reasons include:

- A communication failure exists between the NAS and the LDAP server.
- The LDAP server IP address or port number configured on the NAS is not correct.
- The username is not in the *userid@isp-name* format, or the ISP domain is not correctly configured on the NAS.
- The user is not configured on the LDAP server.
- The password entered by the user is incorrect.
- The administrator DN or password is not configured.
- Some user attributes (for example, the username attribute) configured on the NAS are not consistent with those configured on the server.
- No user search base DN is specified for the LDAP scheme.

Solution

To resolve the problem:

1. Verify the following items:
   - The NAS and the LDAP server can ping each other.
   - The IP address and port number of the LDAP server configured on the NAS match those of the server.
   - The username is in the correct format and the ISP domain for the user authentication is correctly configured on the NAS.
   - The user is configured on the LDAP server.
   - The correct password is entered.
   - The administrator DN and the administrator password are correctly configured.
   - The user attributes (for example, the username attribute) configured on the NAS are consistent with those configured on the LDAP server.

- The user search base DN for authentication is specified.

2. If the problem persists, contact NSFOCUS Support.

Appendixes

Appendix A Commonly used RADIUS attributes

Commonly used RADIUS attributes are defined in RFC 2865, RFC 2866, RFC 2867, and RFC 2868.

Table 5 Commonly used RADIUS attributes

No.	Attribute	No.	Attribute
1	User-Name	45	Acct-Authentic
2	User-Password	46	Acct-Session-Time
3	CHAP-Password	47	Acct-Input-Packets
4	NAS-IP-Address	48	Acct-Output-Packets
5	NAS-Port	49	Acct-Terminate-Cause
6	Service-Type	50	Acct-Multi-Session-Id
7	Framed-Protocol	51	Acct-Link-Count
8	Framed-IP-Address	52	Acct-Input-Gigawords
9	Framed-IP-Netmask	53	Acct-Output-Gigawords
10	Framed-Routing	54	(unassigned)
11	Filter-ID	55	Event-Timestamp
12	Framed-MTU	56-59	(unassigned)
13	Framed-Compression	60	CHAP-Challenge
14	Login-IP-Host	61	NAS-Port-Type
15	Login-Service	62	Port-Limit
16	Login-TCP-Port	63	Login-LAT-Port
17	(unassigned)	64	Tunnel-Type
18	Reply-Message	65	Tunnel-Medium-Type
19	Callback-Number	66	Tunnel-Client-Endpoint
20	Callback-ID	67	Tunnel-Server-Endpoint
21	(unassigned)	68	Acct-Tunnel-Connection
22	Framed-Route	69	Tunnel-Password
23	Framed-IPX-Network	70	ARAP-Password
24	State	71	ARAP-Features
25	Class	72	ARAP-Zone-Access
26	Vendor-Specific	73	ARAP-Security
27	Session-Timeout	74	ARAP-Security-Data
28	Idle-Timeout	75	Password-Retry

No.	Attribute	No.	Attribute
29	Termination-Action	76	Prompt
30	Called-Station-Id	77	Connect-Info
31	Calling-Station-Id	78	Configuration-Token
32	NAS-Identifier	79	EAP-Message
33	Proxy-State	80	Message-Authenticator
34	Login-LAT-Service	81	Tunnel-Private-Group-ID
35	Login-LAT-Node	82	Tunnel-Assignment-id
36	Login-LAT-Group	83	Tunnel-Preference
37	Framed-AppleTalk-Link	84	ARAP-Challenge-Response
38	Framed-AppleTalk-Network	85	Acct-Interim-Interval
39	Framed-AppleTalk-Zone	86	Acct-Tunnel-Packets-Lost
40	Acct-Status-Type	87	NAS-Port-Id
41	Acct-Delay-Time	88	Framed-Pool
42	Acct-Input-Octets	89	(unassigned)
43	Acct-Output-Octets	90	Tunnel-Client-Auth-id
44	Acct-Session-Id	91	Tunnel-Server-Auth-id

Appendix B Descriptions for commonly used standard RADIUS attributes

No.	Attribute	Description
1	User-Name	Name of the user to be authenticated.
2	User-Password	User password for PAP authentication, only present in Access-Request packets when PAP authentication is used.
3	CHAP-Password	Digest of the user password for CHAP authentication, only present in Access-Request packets when CHAP authentication is used.
4	NAS-IP-Address	IP address for the server to use to identify the client. Typically, a client is identified by the IP address of its access interface. This attribute is only present in Access-Request packets.
5	NAS-Port	Physical port of the NAS that the user accesses.
6	Service-Type	Type of service that the user has requested or type of service to be provided.
7	Framed-Protocol	Encapsulation protocol for framed access.
8	Framed-IP-Address	IP address assigned to the user.
11	Filter-ID	Name of the filter list. This attribute is parsed as follows: • If the name is a string of all digits, it indicates an ACL number. • If the name is a string in the format of user-group=*name1*;*name2*;..;*namex*, it indicates a list of user group names. This type of filter list is applicable only to SSL VPN users.

No.	Attribute	Description
		- If the name is not a string of all digits and the name string does not contain an equal sign (=), the device parses this attribute as follows: - If the NSFOCUS-ACL-Version attribute is issued next to this attribute, the value for this attribute is an ACL name. - If the NSFOCUS-ACL-Version attribute is not issued next to this attribute, the value for this attribute is a user profile name.
12	Framed-MTU	MTU for the data link between the user and NAS.
14	Login-IP-Host	IP address of the NAS interface that the user accesses.
15	Login-Service	Type of service that the user uses for login.
18	Reply-Message	Text to be displayed to the user, which can be used by the server to communicate information, for example, the cause of the authentication failure.
26	Vendor-Specific	Vendor-specific proprietary attribute. A packet can contain one or more proprietary attributes, each of which can contain one or more subattributes.
27	Session-Timeout	Maximum service duration for the user before termination of the session.
28	Idle-Timeout	Maximum idle time permitted for the user before termination of the session.
31	Calling-Station-Id	User identification that the NAS sends to the server. For the LAN access service provided by an NSFOCUS device, this attribute includes the MAC address of the user.
32	NAS-Identifier	Identification that the NAS uses to identify itself to the RADIUS server.
40	Acct-Status-Type	Type of the Accounting-Request packet. Possible values include: - **1**—Start. - **2**—Stop. - **3**—Interim-Update. - **4**—Reset-Charge. - **7**—Accounting-On. (Defined in the 3rd Generation Partnership Project.) - **8**—Accounting-Off. (Defined in the 3rd Generation Partnership Project.) - **9 to 14**—Reserved for tunnel accounting. - **15**—Reserved for failed.
45	Acct-Authentic	Authentication method used by the user. Possible values include: - **1**—RADIUS. - **2**—Local. - **3**—Remote.
60	CHAP-Challenge	CHAP challenge generated by the NAS for MD5 calculation during CHAP authentication.
61	NAS-Port-Type	Type of the physical port of the NAS that is authenticating the user. Possible values include: - **15**—Ethernet. - **16**—Any type of ADSL. - **17**—Cable. (With cable for cable TV.) - **19**—WLAN-IEEE 802.11. - **201**—VLAN. - **202**—ATM.

No.	Attribute	Description
		If the port is an ATM or Ethernet one and VLANs are implemented on it, the value of this attribute is 201.
64	Tunnel-Type	Tunneling protocols used. The value **13** represents VLAN. If the value is 13, the device interprets the Tunnel-Type, Tunnel-Medium-Type, and Tunnel-Private-Group-ID attributes as attributes to assign VLANs.
65	Tunnel-Medium-Type	Transport medium type to use for creating a tunnel. For VLAN assignment, the value must be **6** to indicate the 802 media plus Ethernet.
79	EAP-Message	Used to encapsulate EAP packets to allow RADIUS to support EAP authentication.
80	Message-Authenticator	Used for authentication and verification of authentication packets to prevent spoofing Access-Requests. This attribute is present when EAP authentication is used.
81	Tunnel-Private-Group-ID	Group ID for a tunnel session. To assign VLANs, the NAS conveys VLAN IDs by using this attribute.
87	NAS-Port-Id	String for describing the port of the NAS that is authenticating the user.

Appendix C RADIUS subattributes (vendor ID 25506)

Table 6 lists all RADIUS subattributes with a vendor ID of 25506. Support for these subattributes depends on the device model.

Table 6 RADIUS subattributes (vendor ID 25506)

No.	Subattribute	Description
1	Input-Peak-Rate	Peak rate in the direction from the user to the NAS, in bps.
2	Input-Average-Rate	Average rate in the direction from the user to the NAS, in bps.
3	Input-Basic-Rate	Basic rate in the direction from the user to the NAS, in bps.
4	Output-Peak-Rate	Peak rate in the direction from the NAS to the user, in bps.
5	Output-Average-Rate	Average rate in the direction from the NAS to the user, in bps.
6	Output-Basic-Rate	Basic rate in the direction from the NAS to the user, in bps.
15	Remanent_Volume	Total amount of data available for the connection, in different units for different server types.
17	ISP-ID	ISP domain where the user obtains authorization information.
20	Command	Operation for the session, used for session control. Possible values include: • **1**—Trigger-Request. • **2**—Terminate-Request. • **3**—SetPolicy. • **4**—Result. • **5**—PortalClear.
21	ACL-Version	IP protocol version for an ACL. This attribute is used with the Filter-ID attribute to identify the IP protocol version of the ACL in the filter list. • **1**—IPv4.

No.	Subattribute	Description
		• **2**—IPv6.
24	Control_Identifier	Identifier for a packet that is resent by the server. For packets resent by the server during the same session, the value of this attribute is the same. For packets resent by the server during different sessions, the value of this attribute might be the same. Response packets from the corresponding client must carry this attribute with the value the same as the packet resent by the server. This attribute can be ignored in start-accounting, stop-accounting, and interim-update-accounting requests.
25	Result_Code	Result of the Trigger-Request or SetPolicy operation, zero for success and any other value for failure.
26	Connect_ID	Index of the user connection.
27	PortalURL	PADM redirect URL assigned to PPPoE users.
28	Ftp_Directory	FTP, SFTP, or SCP user working directory. When the RADIUS client acts as the FTP, SFTP, or SCP server, this attribute is used to set the working directory for an FTP, SFTP, or SCP user on the RADIUS client.
29	Exec_Privilege	EXEC user priority.
32	NAT-IP-Address	Public IP address assigned to the user when the source IP address and port are translated.
33	NAT-Start-Port	Start port number of the port range assigned to the user when the source IP address and port are translated.
34	NAT-End-Port	End port number of the port range assigned to the user when the source IP address and port are translated.
59	NAS_Startup_Timestamp	Startup time of the NAS in seconds, which is represented by the time elapsed after 00:00:00 on Jan. 1, 1970 (UTC).
60	Ip_Host_Addr	User IP address and MAC address included in authentication and accounting requests, in the format A.B.C.D hh:hh:hh:hh:hh:hh. A space is required between the IP address and the MAC address.
61	User_Notify	Information that must be sent from the server to the client transparently.
98	Multicast_Receive_Group	IP address of the multicast group that the user's host joins as a receiver. This subattribute can appear multiple times in a multicast packet to indicate that the user belongs to multiple multicast groups.
100	IP6_Multicast_Receive_Group	IPv6 address of the multicast group that the user's host joins as a receiver. This subattribute can appear multiple times in a multicast packet to indicate that the user belongs to multiple multicast groups.
101	MLD-Access-Limit	Maximum number of MLD multicast groups that the user can join concurrently.
102	local-name	L2TP local tunnel name.
103	IGMP-Access-Limit	Maximum number of IGMP multicast groups that the user can join concurrently.
104	VPN-Instance	MPLS L3VPN instance to which a user belongs.
105	ANCP-Profile	ANCP profile name.

No.	Subattribute	Description
111	Longitude-Latitude	Longitude and latitude information of the NAS.
135	Client-Primary-DNS	IP address of the primary DNS server.
136	Client-Secondary-DNS	IP address of the secondary DNS server.
140	User_Group	User groups assigned after the user passes authentication. Typically, a user can belong to only one user group. An SSL VPN user can belong to multiple user groups that are separated by semicolons.
141	Security_Level	Security level assigned after the SSL VPN user passes security authentication.
144	Acct_IPv6_Input_Octets	Bytes of IPv6 packets in the inbound direction. The measurement unit depends on the configuration on the device.
145	Acct_IPv6_Output_Octets	Bytes of IPv6 packets in the outbound direction. The measurement unit depends on the configuration on the device.
146	Acct_IPv6_Input_Packets	Number of IPv6 packets in the inbound direction. The measurement unit depends on the configuration on the device.
147	Acct_IPv6_Output_Packets	Number of IPv6 packets in the outbound direction. The measurement unit depends on the configuration on the device.
148	Acct_IPv6_Input_Gigawords	Bytes of IPv6 packets in the inbound direction. The measurement unit is 4G bytes.
149	Acct_IPv6_Output_Gigawords	Bytes of IPv6 packets in the outbound direction. The measurement unit is 4G bytes.
155	User-Roles	List of space-separated user roles.
182	Microsegment-Id	Microsegment ID.
210	Av-Pair	User-defined attribute pair. Available attribute pairs include: • Server-assigned voice VLAN in the format of device-traffic-class=voice. • Server-assigned user role in the format of shell:role=xxx. • Server-deployed command to reboot a port, in the format of subscriber:command=bounce-host-port. • Server-assigned port shutdown duration in the format of bounce:seconds=xxx. • Server-deployed command to shut down a port, in the format of subscriber:command=disable-host-port.
246	Auth_Detail_Result	Accounting details. The server sends Access-Accept packets with subattributes 246 and 250 in the following situations: • **1**—The subscriber charge is overdue. The subscriber is allowed to access network resources in the whitelist. If the subscriber accesses other network resources, the device redirects it to the URL specified by subattribute 250. • **2**—The broadband lease of the subscriber expires. The device redirects the subscriber to the URL specified by subattribute 250 when the subscriber requests to access webpages for the first time.
247	Input-Committed-Burst-Size	Committed burst size from the user to the NAS, in bits. The total length cannot exceed 4 bytes for this field. This subattribute must be assigned together with the Input-Average-Rate attribute.
248	Output-Committed-Burst-Size	Committed burst size from the NAS to the user, in bits. The total

No.	Subattribute	Description
		length cannot exceed 4 bytes for this field. This subattribute must be assigned together with the Output-Average-Rate attribute.
249	authentication-type	Authentication type. The value can be: • **1**—Intranet access authentication. • **2**—Internet access authentication. If the packet does not contain this subattribute, common authentication applies.
250	WEB-URL	Redirect URL for PPP users.
251	Subscriber-ID	Family plan ID.
252	Subscriber-Profile	QoS policy name for the family plan of the subscriber.
255	Product_ID	Product name.

Appendix D HWTACACS attributes

HWTACACS authorization and accounting packets include the HWTACACS attributes assigned by the server to a user and the HWTACACS attributes uploaded by a user to the server. Table 7 shows the HWTACACS attributes supported by the device. The device ignores an HWTACACS attribute if it does not support that attribute.

Table 7 Supported HWTACACS attributes

Attribute name	Description
acl	Number of the ACL assigned to the user.
idletime	Idle timeout period, in seconds.
priv-lvl	User privilege level in the range of level 0 to level 15.
ftp-directory	Initial working directory of the FTP user.
addr	IP address assigned to the user.
addr-pool	IP address pool assigned to the user.
tunnel-type	Type of the tunnel to be established. Only L2TP is supported.
ip-addresses	LNS IP addresses.
tunnel-id	Group ID of the L2TP tunnel.
gw-password	Authentication password of the L2TP tunnel.
roles	User roles assigned to the user.
allowed-roles	Roles for which the user is allowed to obtain temporary authorization.

Contents

802.1X overview

About the 802.1X protocol

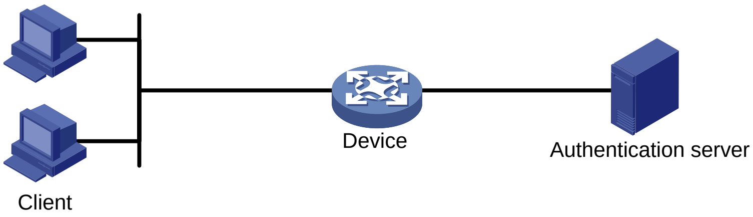
802.1X is a port-based network access control protocol widely used on Ethernet networks. The protocol controls network access by authenticating the devices connected to 802.1X-enabled LAN ports.

802.1X architecture

802.1X operates in the client/server model. As shown in Figure 1, 802.1X authentication includes the following entities:

- **Client (supplicant)**—A user terminal seeking access to the LAN. The terminal must have 802.1X software to authenticate to the access device.
- **Access device (authenticator)**—Authenticates the client to control access to the LAN. In a typical 802.1X environment, the access device uses an authentication server to perform authentication.
- **Authentication server**—Provides authentication services for the access device. The authentication server first authenticates 802.1X clients by using the data sent from the access device. Then, the server returns the authentication results to the access device to make access decisions. The authentication server is typically a RADIUS server. In a small LAN, you can use the access device as the authentication server.

Figure 1 802.1X architecture

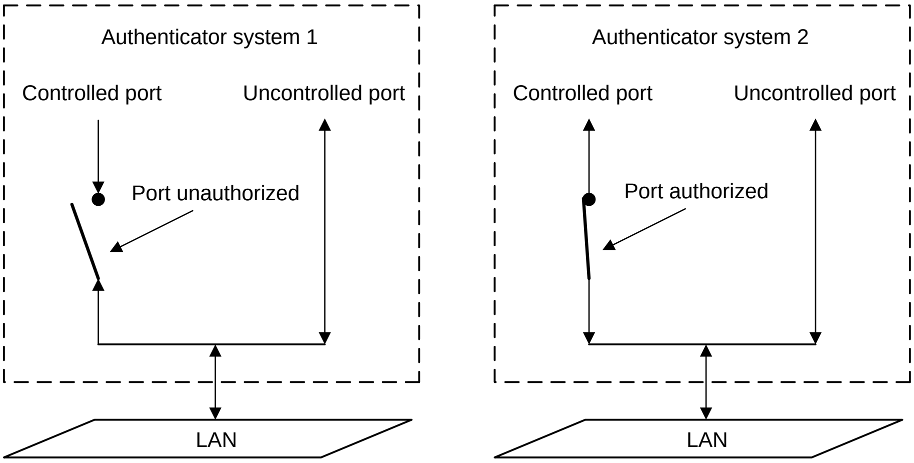


Controlled/uncontrolled port and port authorization status

802.1X defines two logical ports for the network access port: controlled port and uncontrolled port. Any packet arriving at the network access port is visible to both logical ports.

- **Uncontrolled port**—Is always open to receive and transmit authentication packets.
- **Controlled port**—Filters packets depending on the port state.
    - **Authorized state**—The controlled port is in authorized state when the client has passed authentication. The port allows traffic to pass through.
    - **Unauthorized state**—The port is in unauthorized state when the client has failed authentication. The port controls traffic by using one of the following methods:
        - Performs bidirectional traffic control to deny traffic to and from the client.
        - Performs unidirectional traffic control to deny traffic from the client. The device supports only unidirectional traffic control.

Figure 2 Authorization state of a controlled port



Packet exchange methods

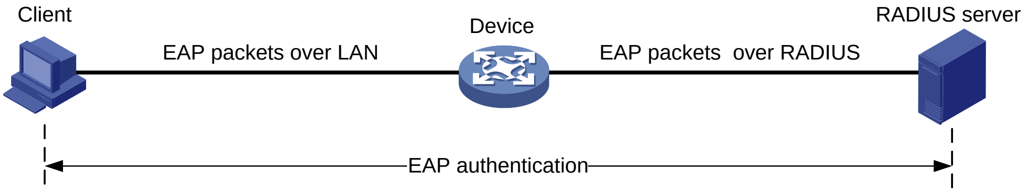
802.1X uses the Extensible Authentication Protocol (EAP) to transport authentication information for the client, the access device, and the authentication server. EAP is an authentication framework that uses the client/server model. The framework supports a variety of authentication methods, including MD5-Challenge, EAP-Transport Layer Security (EAP-TLS), and Protected EAP (PEAP).

802.1X defines EAP over LAN (EAPOL) for passing EAP packets between the client and the access device over a wired or wireless LAN. Between the access device and the authentication server, 802.1X delivers authentication information by either EAP relay or EAP termination.

EAP relay

EAP relay is defined in IEEE 802.1X. In this mode, the network device uses EAP over RADIUS (EAPOR) packets to send authentication information to the RADIUS server, as shown in Figure 3.

Figure 3 EAP relay



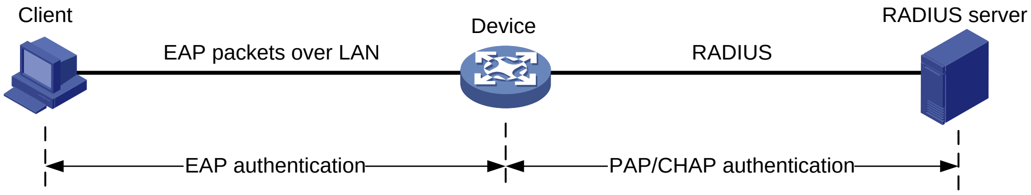
In EAP relay mode, the client must use the same authentication method as the RADIUS server. On the access device, you only need to use the **dot1x authentication-method eap** command to enable EAP relay.

EAP termination

As shown in Figure 4, the access device performs the following operations in EAP termination mode:

1. Terminates the EAP packets received from the client.
2. Encapsulates the client authentication information in standard RADIUS packets.
3. Uses PAP or CHAP to authenticate to the RADIUS server.

Figure 4 EAP termination



Comparing EAP relay and EAP termination

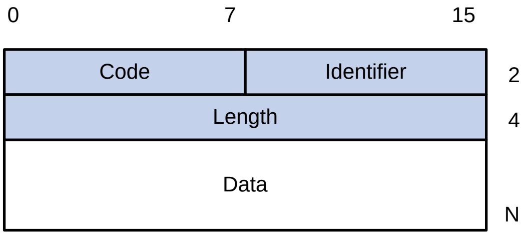
Packet exchange method	Benefits	Limitations
EAP relay	• Supports various EAP authentication methods. • The configuration and processing are simple on the access device.	The RADIUS server must support the EAP-Message and Message-Authenticator attributes, and the EAP authentication method used by the client.
EAP termination	Works with any RADIUS server that supports PAP or CHAP authentication.	• Supports only the following EAP authentication methods: ○ MD5-Challenge EAP authentication. ○ The username and password EAP authentication initiated by an iNode 802.1X client. • The processing is complex on the access device.

Packet formats

EAP packet format

Figure 5 shows the EAP packet format.

Figure 5 EAP packet format

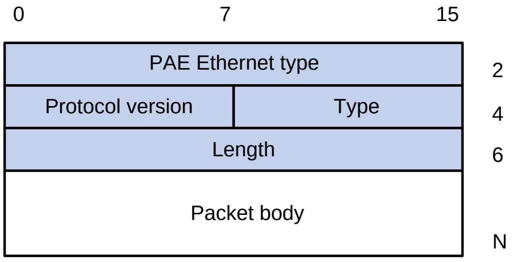


- **Code**—Type of the EAP packet. Options include Request (1), Response (2), Success (3), or Failure (4).
- **Identifier**—Used for matching Responses with Requests.
- **Length**—Length (in bytes) of the EAP packet. The EAP packet length is the sum of the Code, Identifier, Length, and Data fields.
- **Data**—Content of the EAP packet. This field appears only in a Request or Response EAP packet. The **Data** field contains the request type (or the response type) and the type data. Type 1 (Identity) and type 4 (MD5-Challenge) are two examples for the type field.

EAPOL packet format

Figure 6 shows the EAPOL packet format.

Figure 6 EAPOL packet format



- **PAE Ethernet type**—Protocol type. It takes the value 0x888E for EAPOL.
- **Protocol version**—The EAPOL protocol version used by the EAPOL packet sender.
- **Type**—Type of the EAPOL packet. Table 1 lists the types of EAPOL packets supported by the 802.1X implementation of the device.

  Table 1 Types of EAPOL packets

Value	Type	Description
0x00	EAP-Packet	The client and the access device uses EAP-Packets to transport authentication information.
0x01	EAPOL-Start	The client sends an EAPOL-Start message to initiate 802.1X authentication to the access device.
0x02	EAPOL-Logoff	The client sends an EAPOL-Logoff message to tell the access device that the client is logging off.

- **Length**—Data length in bytes, or length of the Packet body. If packet type is EAPOL-Start or EAPOL-Logoff, this field is set to 0, and no Packet body field follows.
- **Packet body**—Content of the packet. When the EAPOL packet type is EAP-Packet, the Packet body field contains an EAP packet.

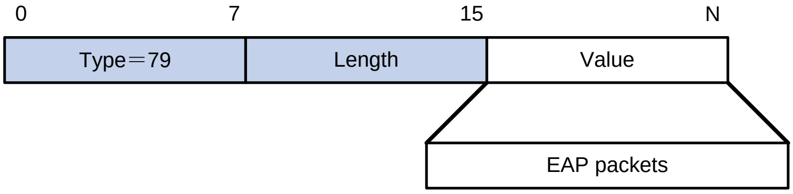
EAP over RADIUS

RADIUS adds two attributes, EAP-Message and Message-Authenticator, for supporting EAP authentication. For more information about the RADIUS packet format, see "Configuring AAA."

- EAP-Message.

  RADIUS encapsulates EAP packets in the EAP-Message attribute, as shown in Figure 7. The Type field takes 79, and the Value field can be up to 253 bytes. If an EAP packet is longer than 253 bytes, RADIUS encapsulates it in multiple EAP-Message attributes.

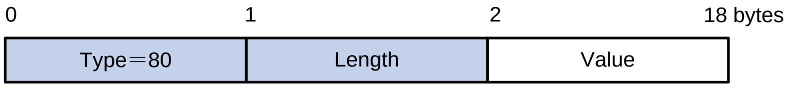
  Figure 7 EAP-Message attribute format



- Message-Authenticator.

  As shown in Figure 8, RADIUS includes the Message-Authenticator attribute in all packets that have an EAP-Message attribute to check their integrity. The packet receiver drops the packet if the calculated packet integrity checksum is different from the Message-Authenticator attribute value. The Message-Authenticator prevents EAP authentication packets from being tampered with during EAP authentication.

Figure 8 Message-Authenticator attribute format



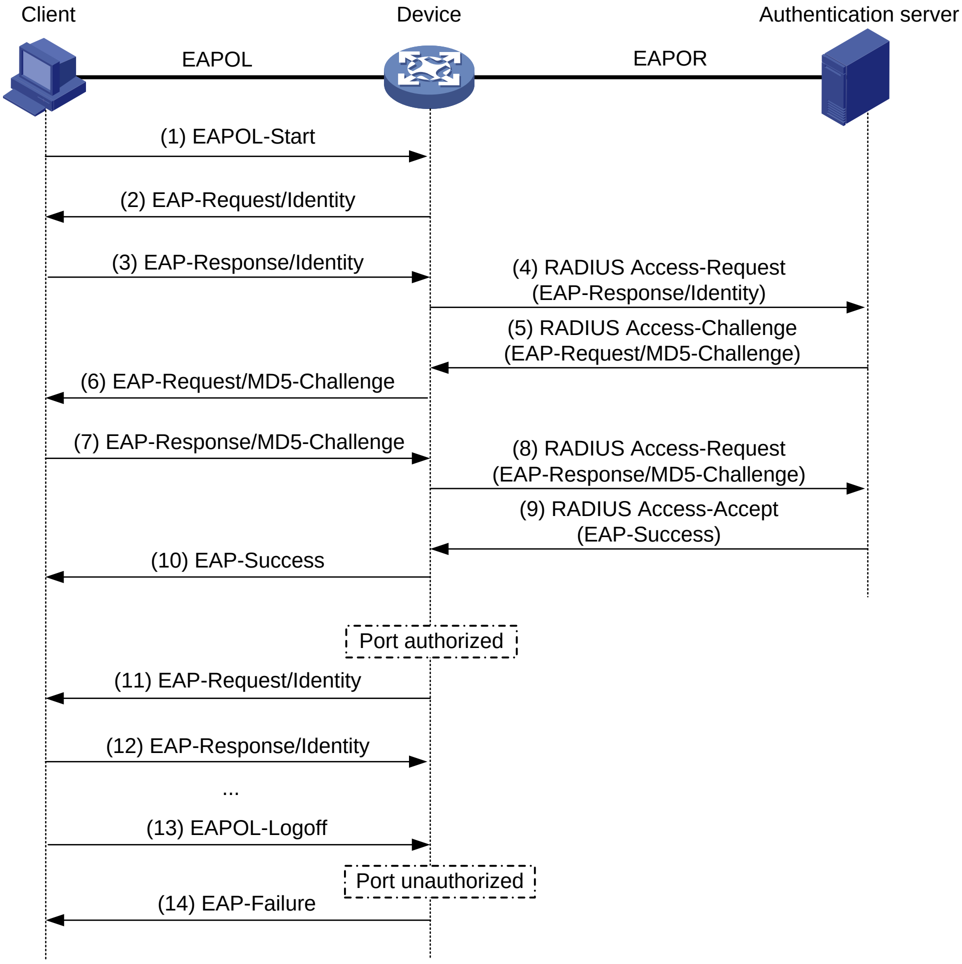
802.1X authentication procedures

802.1X authentication has two methods: EAP relay and EAP termination. You choose either mode depending on support of the RADIUS server for EAP packets and EAP authentication methods.

EAP relay

Figure 9 shows the basic 802.1X authentication procedure in EAP relay mode, assuming that MD5-Challenge EAP authentication is used.

Figure 9 802.1X authentication procedure in EAP relay mode


Client
Device
Authentication server
EAPOL
EAPOR
(1) EAPOL-Start
(2) EAP-Request/Identity
(3) EAP-Response/Identity
(4) RADIUS Access-Request (EAP-Response/Identity)
(5) RADIUS Access-Challenge (EAP-Request/MD5-Challenge)
(6) EAP-Request/MD5-Challenge
(7) EAP-Response/MD5-Challenge
(8) RADIUS Access-Request (EAP-Response/MD5-Challenge)
(9) RADIUS Access-Accept (EAP-Success)
(10) EAP-Success
Port authorized
(11) EAP-Request/Identity
(12) EAP-Response/Identity
...
(13) EAPOL-Logoff
Port unauthorized
(14) EAP-Failure

The following steps describe the 802.1X authentication procedure:

1. When a user launches the 802.1X client and enters a registered username and password, the 802.1X client sends an EAPOL-Start packet to the access device.
2. The access device responds with an EAP-Request/Identity packet to ask for the client username.
3. In response to the EAP-Request/Identity packet, the client sends the username in an EAP-Response/Identity packet to the access device.

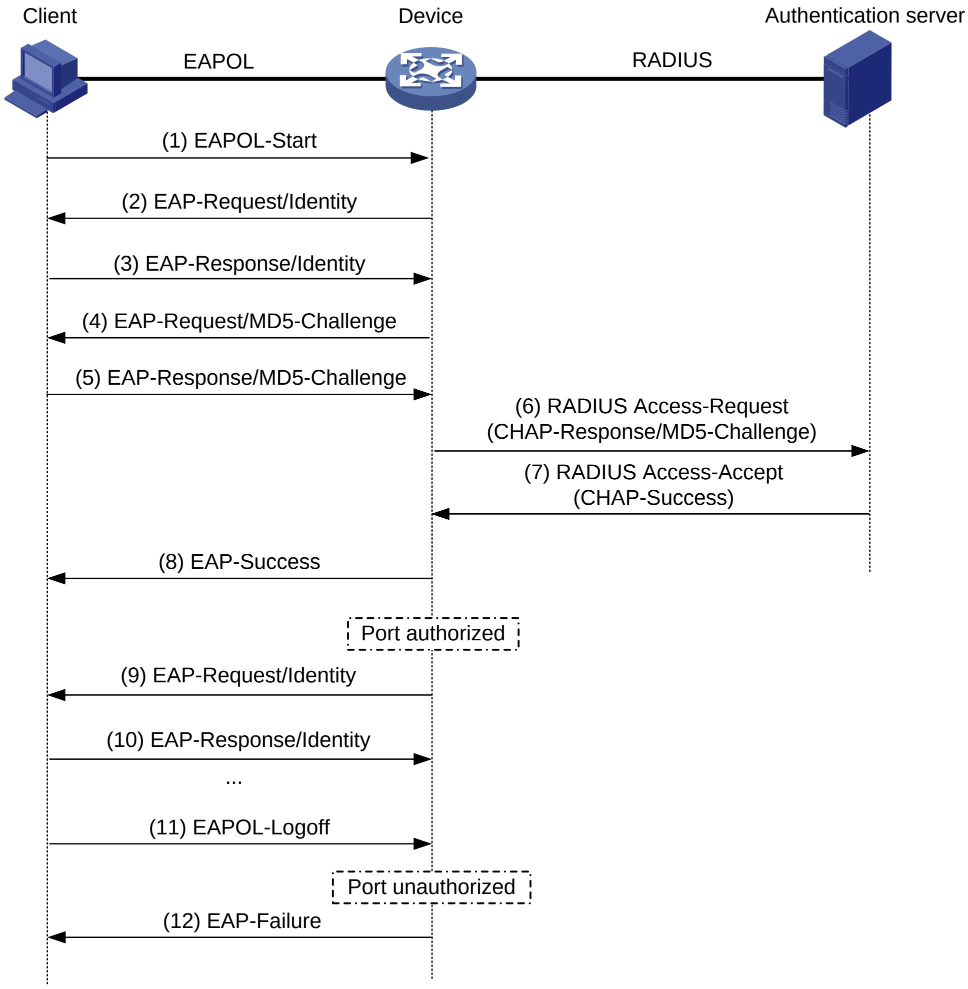
4. The access device relays the EAP-Response/Identity packet in a RADIUS Access-Request packet to the authentication server.
5. The authentication server uses the identity information in the RADIUS Access-Request to search its user database. If a matching entry is found, the server uses a randomly generated challenge (EAP-Request/MD5-Challenge) to encrypt the password in the entry. Then, the server sends the challenge in a RADIUS Access-Challenge packet to the access device.
6. The access device transmits the EAP-Request/MD5-Challenge packet to the client.
7. The client uses the received challenge to encrypt the password, and sends the encrypted password in an EAP-Response/MD5-Challenge packet to the access device.
8. The access device relays the EAP-Response/MD5-Challenge packet in a RADIUS Access-Request packet to the authentication server.
9. The authentication server compares the received encrypted password with the encrypted password it generated at step 5. If the two passwords are identical, the server considers the client valid and sends a RADIUS Access-Accept packet to the access device.
10. Upon receiving the RADIUS Access-Accept packet, the access device performs the following operations:
    a. Sends an EAP-Success packet to the client.
    b. Sets the controlled port in authorized state.

    The client can access the network.
11. After the client comes online, the access device periodically sends handshake requests to check whether the client is still online. By default, if two consecutive handshake attempts fail, the device logs off the client.
12. Upon receiving a handshake request, the client returns a response. If the client fails to return a response after a number of consecutive handshake attempts (two by default), the access device logs off the client. This handshake mechanism enables timely release of the network resources used by 802.1X users that have abnormally gone offline.
13. The client can also send an EAPOL-Logoff packet to ask the access device for a logoff.
14. In response to the EAPOL-Logoff packet, the access device changes the status of the controlled port from authorized to unauthorized. Then, the access device sends an EAP-Failure packet to the client.

EAP termination

Figure 10 shows the basic 802.1X authentication procedure in EAP termination mode, assuming that CHAP authentication is used.

Figure 10 802.1X authentication procedure in EAP termination mode



In EAP termination mode, the access device rather than the authentication server generates an MD5 challenge for password encryption. The access device then sends the MD5 challenge together with the username and encrypted password in a standard RADIUS packet to the RADIUS server.

802.1X authentication initiation

Both the 802.1X client and the access device can initiate 802.1X authentication.

802.1X client as the initiator

The client sends an EAPOL-Start packet to the access device to initiate 802.1X authentication. The destination MAC address of the packet is the IEEE 802.1X specified multicast address 01-80-C2-00-00-03 or the broadcast MAC address. If any intermediate device between the client and the authentication server does not support the multicast address, you must use an 802.1X client that can send broadcast EAPOL-Start packets. For example, you can use the iNode 802.1X client.

Access device as the initiator

If the client cannot send EAPOL-Start packets, configure the access device to initiate authentication.

The access device supports the following modes:

- **Multicast trigger mode**—The access device multicasts EAP-Request/Identity packets to initiate 802.1X authentication at the identity request interval.

- **Unicast trigger mode**—Upon receiving a frame from an unknown MAC address, the access device sends an EAP-Request/Identity packet out of the receiving port to the MAC address. The device retransmits the packet if no response has been received within the identity request timeout interval. This process continues until the maximum number of request attempts set by using the **`dot1x retry`** command is reached.

The username request timeout timer sets both the identity request interval for the multicast trigger and the identity request timeout interval for the unicast trigger.

Access control methods

NSFOCUS implements port-based access control as defined in the 802.1X protocol, and extends the protocol to support MAC-based access control.

- **Port-based access control**—Once an 802.1X user passes authentication on a port, any subsequent user can access the network through the port without authentication. When the authenticated user logs off, all other users are logged off.
- **MAC-based access control**—Each user is separately authenticated on a port. When a user logs off, no other online users are affected.

802.1X VLAN manipulation

Authorization VLAN

The authorization VLAN controls the access of an 802.1X user to authorized network resources. The device supports authorization VLANs assigned locally or by a remote server.

ⓘ **IMPORTANT:**

Only remote servers can assign tagged authorization VLANs.

Remote VLAN authorization

In remote VLAN authorization, you must configure an authorization VLAN for a user on the remote server. After the user authenticates to the server, the server assigns authorization VLAN information to the device. Then, the device assigns the user access port to the authorization VLAN as a tagged or untagged member.

The device supports assignment of the following authorization VLAN information by the remote server:

- VLAN ID.
- VLAN name, which must be the same as the VLAN description on the access device.
- A string of VLAN IDs and VLAN names.

  In the string, some VLANs are represented by their IDs, and some VLANs are represented by their names.
- VLAN group name.

  For more information about VLAN groups, see VLAN configuration in *Layer 2—LAN Switching Configuration Guide*.
- VLAN ID with a suffix of **t** or **u**.

  The **t** and **u** suffixes require the device to assign the access port to the VLAN as a tagged or untagged member, respectively. For example, **2u** indicates assigning the port to VLAN 2 as an untagged member.

If a VLAN name or VLAN group name is assigned, the device converts the information into a VLAN ID before VLAN assignment.

IMPORTANT:

For a VLAN represented by its VLAN name to be assigned successfully, you must make sure the VLAN has been created on the device.

To assign VLAN IDs with suffixes, make sure the user access port is a hybrid or trunk port that performs port-based access control.

To ensure a successful assignment, the authorization VLANs assigned by the remote server cannot be any of the following types:

- Dynamically learned VLANs.
- Reserved VLANs.

If the server assigns a group of VLANs, the access device selects a VLAN as described in Table 2.

Table 2 Authorization VLAN selection from a group of VLANs

VLAN information	Authorization VLAN selection
VLANs by IDs VLANs by names VLAN group name	If the 802.1X-enabled port performs port-based access control, the device selects the VLAN with the lowest ID from the VLAN group. All subsequent 802.1X users are assigned to that VLAN.
VLAN IDs with suffixes	**1.** The device selects the leftmost VLAN ID without a suffix, or the leftmost VLAN ID suffixed by **u** as an untagged VLAN, whichever is more leftmost. **2.** The device assigns the untagged VLAN to the port as the PVID, and it assigns the remaining as tagged VLANs. If no untagged VLAN is assigned, the PVID of the port does not change. The port permits traffic from these tagged and untagged VLANs to pass through. For example, the authentication server sends the string **1u 2t 3** to the access device for a user. The device assigns VLAN 1 as an untagged VLAN and all remaining VLANs (including VLAN 3) as tagged VLANs. VLAN 1 becomes the PVID.

Local VLAN authorization

To perform local VLAN authorization for a user, specify the VLAN ID in the authorization attribute list of the local user account for that user. For each local user, you can specify only one authorization VLAN ID. The user access port is assigned to the VLAN as an untagged member.

IMPORTANT:

Local VLAN authorization does not support assignment of tagged VLANs.

For more information about local user configuration, see "Configuring AAA."

Authorization VLAN manipulation on an 802.1X-enabled port

Table 3 describes how the access device handles VLANs (except for the VLANs specified with suffixes) on an 802.1X-enabled port.

Table 3 VLAN manipulation

Port access control method	VLAN manipulation
Port-based	The device assigns the port to the first authenticated user's authorization VLAN. All subsequent 802.1X users can access the VLAN without authentication.

Port access control method	VLAN manipulation
	If the authorization VLAN has the untagged attribute, the device assigns the port to the authorization VLAN as an untagged member and sets the VLAN as the PVID. If the authorization VLAN has the tagged attribute, the device assigns the port to the VLAN as a tagged member without changing the PVID. **NOTE:** The tagged attribute is supported only on trunk and hybrid ports.
MAC-based	On a hybrid port with MAC-based VLAN enabled, the device maps the MAC address of each user to its own authorization VLAN. The PVID of the port does not change. On an access, trunk or MAC-based VLAN disabled hybrid port: • The device assigns the port to the first authenticated user's authorization VLAN and sets the VLAN as the PVID if that authorization VLAN has the untagged attribute. • If the authorization VLAN has the tagged attribute, the device assigns the port to the authorization VLAN without changing its PVID.

IMPORTANT:

- If the users are attached to a port whose link type is access, make sure the authorization VLAN assigned by the server has the untagged attribute. VLAN assignment will fail if the server issues a VLAN that has the tagged attribute.
- When you assign VLANs to users attached to a trunk port or a MAC-based VLAN disabled hybrid port, make sure there is only one untagged VLAN. If a different untagged VLAN is assigned to a subsequent user, the user cannot pass authentication.
- As a best practice to enhance network security, do not use the **port hybrid vlan** command to assign a hybrid port to an authorization VLAN as a tagged member.

For an 802.1X authenticated user to access the network on a hybrid port when no authorization VLAN is configured for the user, perform one of the following tasks:

- If the port receives tagged authentication packets from the user in a VLAN, use the **port hybrid vlan** command to configure the port as a tagged member in the VLAN.
- If the port receives untagged authentication packets from the user in a VLAN, use the **port hybrid vlan** command to configure the port as an untagged member in the VLAN.

On a port with periodic online user reauthentication enabled, the MAC-based VLAN feature does not take effect on a user that has been online since before this feature was enabled. The access device creates a MAC-to-VLAN mapping for the user when the following requirements are met:

- The user passes reauthentication.
- The authorization VLAN for the user is changed.

For more information about VLAN configuration and MAC-based VLANs, see *Layer 2—LAN Switching Configuration Guide*.

Guest VLAN

The 802.1X guest VLAN on a port accommodates users that have not performed 802.1X authentication. Users in the guest VLAN can access a limited set of network resources, such as a software server, to download antivirus software and system patches. Once a user in the guest VLAN passes 802.1X authentication, it is removed from the guest VLAN and can access authorized network resources.

The access device handles VLANs on an 802.1X-enabled port based on its 802.1X access control method.

Port-based access control

Authentication status	VLAN manipulation
A user accesses the 802.1X-enabled port when the port is in auto state.	The device assigns the port to the 802.1X guest VLAN. All 802.1X users on this port can access only resources in the guest VLAN. The guest VLAN assignment varies by port link mode. For more information, see Table 3 in "Authorization VLAN."
A user in the 802.1X guest VLAN fails 802.1X authentication.	If an 802.1X Auth-Fail VLAN is available, the device assigns the port to the Auth-Fail VLAN. All users on this port can access only resources in the Auth-Fail VLAN. If no Auth-Fail VLAN is configured, the port is still in the 802.1X guest VLAN. All users on the port are in the guest VLAN. For information about the 802.1X Auth-Fail VLAN, see "Auth-Fail VLAN."
A user in the 802.1X guest VLAN passes 802.1X authentication.	The device removes the port from the 802.1X guest VLAN and assigns the port to the authorization VLAN of the user. If the authentication server does not assign an authorization VLAN, the initial PVID of the port applies. The user and all subsequent 802.1X users are assigned to the initial port VLAN. After the user logs off, the port is assigned to the guest VLAN again. **NOTE:** The initial PVID of an 802.1X-enabled port refers to the PVID used by the port before the port is assigned to any 802.1X VLANs.

MAC-based access control

Authentication status	VLAN manipulation
A user accesses the 802.1X-enabled port and has not performed 802.1X authentication.	The device creates a mapping between the MAC address of the user and the 802.1X guest VLAN. The user can access only resources in the guest VLAN.
A user in the 802.1X guest VLAN fails 802.1X authentication.	If an 802.1X Auth-Fail VLAN is available, the device remaps the MAC address of the user to the Auth-Fail VLAN. The user can access only resources in the Auth-Fail VLAN. If no 802.1X Auth-Fail VLAN is configured, the user is removed from the guest VLAN and added to the initial PVID.
A user in the 802.1X guest VLAN passes 802.1X authentication.	The device remaps the MAC address of the user to the authorization VLAN. If the authentication server does not assign an authorization VLAN, the device remaps the MAC address of the user to the initial PVID on the port.

Auth-Fail VLAN

The 802.1X Auth-Fail VLAN on a port accommodates users that have failed 802.1X authentication because of the failure to comply with the organization security strategy. For example, the VLAN accommodates users that have entered a wrong password. Users in the Auth-Fail VLAN can access a limited set of network resources, such as a software server, to download antivirus software and system patches.

The access device handles VLANs on an 802.1X-enabled port based on its 802.1X access control method.

Port-based access control

Authentication status	VLAN manipulation
A user accesses the port and fails 802.1X authentication.	The device assigns the port to the Auth-Fail VLAN. All 802.1X users on this port can access only resources in the Auth-Fail VLAN. The Auth-Fail VLAN assignment varies by port link mode. For more information, see Table 3 in "Authorization VLAN."
A user in the 802.1X Auth-Fail VLAN fails 802.1X authentication.	The port is still in the Auth-Fail VLAN, and all 802.1X users on this port are in this VLAN.
A user in the 802.1X Auth-Fail VLAN passes 802.1X authentication.	The device assigns the port to the authorization VLAN of the user, and it removes the port from the Auth-Fail VLAN. If the authentication server does not assign an authorization VLAN, the initial PVID of the port applies. The user and all subsequent 802.1X users are assigned to the initial PVID. After the user logs off, the port is assigned to the guest VLAN. If no guest VLAN is configured, the port is assigned to the initial PVID of the port.

MAC-based access control

Authentication status	VLAN manipulation
A user accesses the port and fails 802.1X authentication.	The device maps the MAC address of the user to the 802.1X Auth-Fail VLAN. The user can access only resources in the Auth-Fail VLAN.
A user in the 802.1X Auth-Fail VLAN fails 802.1X authentication.	The user is still in the Auth-Fail VLAN.
A user in the 802.1X Auth-Fail VLAN passes 802.1X authentication.	The device remaps the MAC address of the user to the authorization VLAN. If the authentication server does not assign an authorization VLAN, the device remaps the MAC address of the user to the initial PVID on the port.

Critical VLAN

The 802.1X critical VLAN on a port accommodates 802.1X users that have failed authentication because none of the RADIUS servers in their ISP domain are reachable. Users in the critical VLAN can access a limited set of network resources depending on the configuration.

The critical VLAN feature takes effect when 802.1X authentication is performed only through RADIUS servers. If an 802.1X user fails local authentication after RADIUS authentication, the user is not assigned to the critical VLAN. For more information about the authentication methods, see "Configuring AAA."

The access device handles VLANs on an 802.1X-enabled port based on its 802.1X access control method.

Port-based access control

Authentication status	VLAN manipulation
A user accesses the port and fails 802.1X authentication because all the RADIUS servers are unreachable.	The device assigns the port to the critical VLAN. The 802.1X user and all subsequent 802.1X users on this port can access only resources in the 802.1X critical VLAN. The critical VLAN assignment varies by port link mode. For more information, see Table 3 in "Authorization VLAN."
A user in the 802.1X critical VLAN fails authentication because all the RADIUS servers are unreachable.	The port is still in the critical VLAN.
A user in the 802.1X critical VLAN fails authentication for any reason other than unreachable servers.	If an 802.1X Auth-Fail VLAN has been configured, the port is assigned to the Auth-Fail VLAN. If no 802.1X Auth-Fail VLAN is configured, the port is assigned to the initial PVID of the port.
A user in the 802.1X critical VLAN passes 802.1X authentication.	The device assigns the port to the authorization VLAN of the user, and it removes the port from the 802.1X critical VLAN. If the authentication server does not assign an authorization VLAN, the initial PVID of the port applies. The user and all subsequent 802.1X users are assigned to this port VLAN. After the user logs off, the port is assigned to the guest VLAN. If no 802.1X guest VLAN is configured, the initial PVID of the port is restored.
A user in the 802.1X guest VLAN fails authentication because all the RADIUS servers are unreachable.	The device assigns the port to the 802.1X critical VLAN, and all 802.1X users on this port are in this VLAN.
A user in the 802.1X Auth-Fail VLAN fails authentication because all the RADIUS servers are unreachable.	The port is still in the 802.1X Auth-Fail VLAN. All 802.1X users on this port can access only resources in the 802.1X Auth-Fail VLAN.
A user that has passed authentication fails reauthentication because all the RADIUS servers are unreachable, and the user is logged out of the device.	The device assigns the port to the 802.1X critical VLAN.

MAC-based access control

Authentication status	VLAN manipulation
A user accesses the port and fails 802.1X authentication because all the RADIUS servers are unreachable.	The device maps the MAC address of the user to the 802.1X critical VLAN. The user can access only resources in the 802.1X critical VLAN.
A user in the 802.1X critical VLAN fails authentication because all the RADIUS servers are unreachable.	The user is still in the critical VLAN.
A user in the 802.1X critical VLAN fails 802.1X authentication for any reason other than unreachable servers.	If an 802.1X Auth-Fail VLAN has been configured, the device remaps the MAC address of the user to the Auth-Fail VLAN ID. If no 802.1X Auth-Fail VLAN has been configured, the device remaps the MAC address of the user to the initial PVID.

Authentication status	VLAN manipulation
A user in the 802.1X critical VLAN passes 802.1X authentication.	The device remaps the MAC address of the user to the authorization VLAN. If the authentication server does not assign an authorization VLAN to the user, the device remaps the MAC address of the user to the initial PVID on the port.
A user in the 802.1X guest VLAN fails authentication because all the RADIUS servers are unreachable.	The device remaps the MAC address of the user to the 802.1X critical VLAN. The user can access only resources in the 802.1X critical VLAN.
A user in the 802.1X Auth-Fail VLAN fails authentication because all the RADIUS servers are unreachable.	The user remains in the 802.1X Auth-Fail VLAN.

ACL assignment

You can specify an ACL for an 802.1X user on the authentication server to control the user's access to network resources. After the user passes 802.1X authentication, the authentication server assigns the ACL to the user access port. Then, the port permits or drops the matching traffic from the user depending on the rules configured in the ACL.

The authentication server can be the local access device or a RADIUS server. In either case, the server only specifies the ACL number. You must configure the ACL and its rules on the access device.

To change the access control criteria for the user, you can use one of the following methods:

- Modify ACL rules on the access device.
- Specify another authorization ACL on the authentication server.

The supported authorization ACLs include the following types:

- Basic ACLs, which are numbered in the range of 2000 to 2999.
- Advanced ACLs, which are numbered in the range of 3000 to 3999.
- Layer 2 ACLs, which are numbered in the range of 4000 to 4999.

For an authorization ACL to take effect, make sure the ACL exists with rules and none of the rules contains the **counting**, **established**, **fragment**, **source-mac**, or **logging** keyword.

For more information about ACLs, see *ACL and QoS Command Reference*.

Periodic 802.1X reauthentication

Periodic 802.1X reauthentication tracks the connection status of online users and updates the authorization attributes (such as ACL and VLAN) assigned by the server.

The device reauthenticates online 802.1X users at the periodic reauthentication interval when the periodic online user reauthentication feature is enabled. The interval is controlled by a timer and the timer is user configurable. A change to the periodic reauthentication timer applies to online users only after the old timer expires and the users pass authentication.

The server-assigned session timeout timer (Session-Timeout attribute) and termination action (Termination-Action attribute) together can affect the periodic online user reauthentication feature. To display the server-assigned Session-Timeout and Termination-Action attributes, use the **display dot1x connection** command (see *Security Command Reference*).

- If the termination action is **Default** (logoff), periodic online user reauthentication on the device takes effect only when the periodic reauthentication timer is shorter than the session timeout timer.
- If the termination action is **Radius-request**, the periodic online user reauthentication settings on the device do not take effect. The device reauthenticates the online 802.1X users after the session timeout timer expires.

If no session timeout timer is assigned by the server, whether the device performs periodic 802.1X reauthentication depends on the periodic reauthentication configuration on the device. Support for the assignment of Session-Timeout and Termination-Action attributes depends on the server model.

By default, the device logs off online 802.1X users if no server is reachable for 802.1X reauthentication. The keep-online feature keeps authenticated 802.1X users online when no server is reachable for 802.1X reauthentication.

The VLANs assigned to an online user before and after reauthentication can be the same or different.

EAD assistant

Endpoint Admission Defense (EAD) is an NSFOCUS integrated endpoint access control solution to improve the threat defensive capability of a network. The solution enables the security client, security policy server, access device, and third-party server to operate together. If a terminal device seeks to access an EAD network, it must have an EAD client, which performs 802.1X authentication.

The EAD assistant feature enables the access device to redirect the HTTP requests of a user to a redirect URL for downloading and installing an EAD client. This feature eliminates the administrative task to deploy EAD clients.

EAD assistant is implemented by the following functionality:

- Free IP.

  A free IP is a freely accessible network segment, which has a limited set of network resources such as software and DHCP servers. To ensure security strategy compliance, an unauthenticated user can access only this segment to perform operations. For example, the user can download EAD client from a software server or obtain a dynamic IP address from a DHCP server.
- Redirect URL.

  If an unauthenticated 802.1X user is using a Web browser to access the network, EAD assistant redirects the network access requests of the user to a specific URL. For example, you can use this feature to redirect the user to the EAD client software download page.

The EAD assistant feature creates an ACL-based EAD rule automatically to open access to the redirect URL for each redirected user.

EAD rules are implemented by using ACL resources. When the EAD rule timer expires or the user passes authentication, the rule is removed. If users fail to download EAD client or fail to pass authentication before the timer expires, they must reconnect to the network to access the free IP.

SmartOn

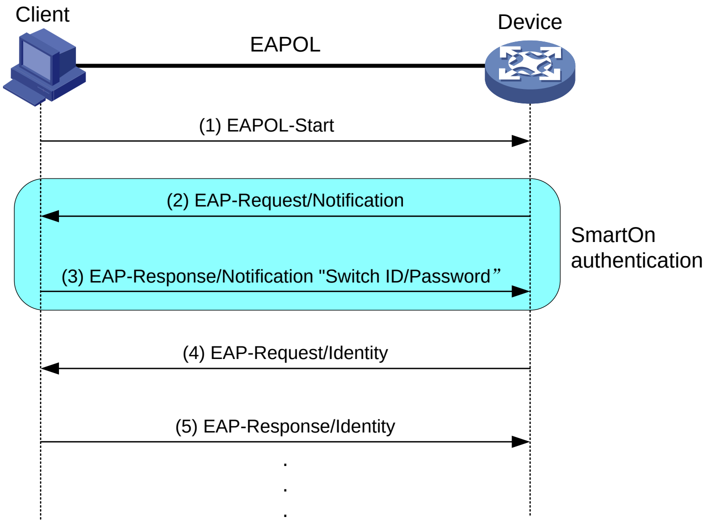
The SmartOn feature was developed to support the NEC 802.1X client.

As shown in Figure 11, the access device performs SmartOn authentication before 802.1X authentication. The following shows the authentication process:

1. When a SmartOn-enabled port receives an EAPOL-Start packet from an 802.1X client, it sends a unicast EAP-Request/Notification packet to the client for SmartOn authentication.

2. Upon receiving an EAP-Response/Notification from the client, the device compares the switch ID and password in the packet with the switch ID and password configured on the device.
   - If they are the same, 802.1X authentication can continue.
   - If they do not match, SmartOn authentication fails. The access device stops 802.1X authentication for the client.

Figure 11 802.1X authentication process with the SmartOn feature


Client
Device
EAPOL
(1) EAPOL-Start
(2) EAP-Request/Notification
(3) EAP-Response/Notification "Switch ID/Password"
SmartOn authentication
(4) EAP-Request/Identity
(5) EAP-Response/Identity

If the user attempts to use another 802.1X client for authentication, it will fail SmartOn authentication. The access device stops 802.1X authentication for the user.

NOTE:

After you install the SmartOn client software, add two values **QX_ID** and **QX_PASSWORD** to the Windows registry key [HKEY_LOCAL_MACHINE\SOFTWARE\Soliton Systems K.K.\SmartOn Client\Clients\1XGate]. Specify the switch ID and password for the QX_ID and QX_PASSWORD, respectively. The switch ID and password must be the same as the switch ID and password configured on the device.

Configuring 802.1X

Restrictions and guidelines: 802.1X configuration

Do not change the link type of a port when the 802.1X guest VLAN, Auth-Fail VLAN, or critical VLAN on the port has users.

802.1X tasks at a glance

To configure 802.1X authentication, perform the following tasks:

1. Enabling 802.1X
2. Configuring basic 802.1X features
   - Enabling EAP relay or EAP termination
   - Setting the port authorization state
   - Specifying an access control method
   - (Optional.) Specifying a mandatory authentication domain on a port
   - (Optional.) Setting the 802.1X authentication timeout timers
   - (Optional.) Configuring 802.1X reauthentication
   - (Optional.) Setting the quiet timer
3. (Optional.) Configuring 802.1X VLAN assignment
   - Configuring an 802.1X guest VLAN
   - Configuring an 802.1X Auth-Fail VLAN
   - Configuring an 802.1X critical VLAN
4. (Optional.) Configuring other 802.1X features
   - Configuring the authentication trigger feature

     Perform this task when 802.1X clients cannot initiate authentication.
   - Setting the maximum number of concurrent 802.1X users on a port
   - Setting the maximum number of authentication request attempts
   - Configuring online user handshake
   - Specifying supported domain name delimiters
   - Configuring the EAD assistant feature
   - Configuring 802.1X SmartOn

Prerequisites for 802.1X

Before you configure 802.1X, complete the following tasks:

- Configure an ISP domain and AAA scheme (local or RADIUS authentication) for 802.1X users.
- If RADIUS authentication is used, create user accounts on the RADIUS server.
- If local authentication is used, create local user accounts on the access device and set the service type to **lan-access**.

Enabling 802.1X

Restrictions and guidelines

For 802.1X to take effect on a port, you must enable it both globally and on the port.

Do not enable 802.1X on a port that is in a link aggregation group.

Procedure

1. Enter system view.

   `system-view`

2. Enable 802.1X globally.

   `dot1x`

   By default, 802.1X is disabled globally.

3. Enter interface view.

   `interface` *`interface-type interface-number`*

4. Enable 802.1X on a port.

   `dot1x`

   By default, 802.1X is disabled on a port.

Enabling EAP relay or EAP termination

About this task

Consider the following factors to select a proper EAP mode:

- Support of the RADIUS server for EAP packets.
- Authentication methods supported by the 802.1X client and the RADIUS server.

Restrictions and guidelines

- If EAP relay mode is used, the **`user-name-format`** command configured in RADIUS scheme view does not take effect. The access device sends the authentication data from the client to the server without any modification. For more information about the **`user-name-format`** command, see AAA commands in *Security Command Reference*.
- You can use both EAP termination and EAP relay in any of the following situations:
  - The client is using only MD5-Challenge EAP authentication. If EAP termination is used, you must enable CHAP authentication on the access device.
  - The client is an iNode 802.1X client and initiates only the username and password EAP authentication. If EAP termination is used, you can enable either PAP or CHAP authentication on the access device. However, for the purpose of security, you must use CHAP authentication on the access device.
- To use EAP-TLS, PEAP, or any other EAP authentication methods, you must use EAP relay. When you make your decision, see "Comparing EAP relay and EAP termination" for help.

Procedure

1. Enter system view.

   `system-view`

2. Configure EAP relay or EAP termination.

   `dot1x authentication-method` { **`chap`** | **`eap`** | **`pap`** }

   By default, the access device performs EAP termination and uses CHAP to communicate with the RADIUS server.

Setting the port authorization state

About this task

The port authorization state determines whether the client is granted access to the network. You can control the following authorization states of a port:

- **Authorized**—Places the port in the authorized state, enabling users on the port to access the network without authentication.
- **Unauthorized**—Places the port in the unauthorized state, denying any access requests from users on the port.
- **Auto**—Places the port initially in unauthorized state to allow only EAPOL packets to pass. After a user passes authentication, sets the port in the authorized state to allow access to the network. You can use this option in most scenarios.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the port authorization state.

   dot1x port-control { **authorized-force** | **auto** | **unauthorized-force** }

   By default, the **auto** state applies.

Specifying an access control method

About this task

The device supports port-based and MAC-based access control methods.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify an access control method.

   dot1x port-method { **macbased** | **portbased** }

   By default, MAC-based access control applies.

△ **CAUTION:**

If online 802.1X users are present on a port, changing its access control method will cause the online users to go offline.

Specifying a mandatory authentication domain on a port

About this task

You can place all 802.1X users in a mandatory authentication domain for authentication, authorization, and accounting on a port. No user can use an account in any other domain to access the network through the port. The implementation of a mandatory authentication domain enhances the flexibility of 802.1X access control deployment.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify a mandatory 802.1X authentication domain on the port.

   `dot1x mandatory-domain` *`domain-name`*

   By default, no mandatory 802.1X authentication domain is specified.

Setting the 802.1X authentication timeout timers

About this task

The network device uses the following 802.1X authentication timeout timers:

- **Client timeout timer**—Starts when the access device sends an EAP-Request/MD5-Challenge packet to a client. If no response is received when this timer expires, the access device retransmits the request to the client.
- **Server timeout timer**—Starts when the access device sends a RADIUS Access-Request packet to the authentication server. If no response is received when this timer expires, 802.1X authentication fails.

Restrictions and guidelines

In most cases, the default settings are sufficient. You can edit the timers, depending on the network conditions.

- In a low-speed network, increase the client timeout timer.
- In a network with authentication servers of different performance, adjust the server timeout timer.

Procedure

1. Enter system view.

   `system-view`

2. Set the client timeout timer.

   `dot1x timer supp-timeout` *`supp-timeout-value`*

   The default is 30 seconds.

3. Set the server timeout timer.

   `dot1x timer server-timeout` *`server-timeout-value`*

   The default is 100 seconds.

Configuring 802.1X reauthentication

Restrictions and guidelines

Any modification to the mandatory authentication domain or EAP message handling method setting does not affect the reauthentication of online 802.1X users. The modified setting takes effect only on 802.1X users that come online after the modification.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the periodic reauthentication timer.

   dot1x timer reauth-period *reauth-period-value*

   The default setting is 3600 seconds.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable periodic online user reauthentication.

   dot1x re-authenticate

   By default, the feature is disabled.

5. (Optional.) Enable the keep-online feature for 802.1X users.

   dot1x re-authenticate server-unreachable keep-online

   By default, this feature is disabled. The device logs off online 802.1X users if no authentication server is reachable for 802.1X reauthentication.

   Use the keep-online feature according to the actual network condition. In a fast-recovery network, you can use the keep-online feature to prevent 802.1X users from coming online and going offline frequently.

Setting the quiet timer

About this task

The quiet timer enables the access device to wait a period of time before it can process any authentication request from a client that has failed an 802.1X authentication.

Restrictions and guidelines

You can edit the quiet timer, depending on the network conditions.

- In a vulnerable network, set the quiet timer to a high value.
- In a high-performance network with quick authentication response, set the quiet timer to a low value.

Procedure

1. Enter system view.

   system-view

2. Enable the quiet timer.

   dot1x quiet-period

   By default, the timer is disabled.

3. (Optional.) Set the quiet timer.

   dot1x timer quiet-period *quiet-period-value*

   The default is 60 seconds.

Configuring an 802.1X guest VLAN

Restrictions and guidelines

- You can configure only one 802.1X guest VLAN on a port. The 802.1X guest VLANs on different ports can be different.
- Assign different IDs to the port VLAN, the voice VLAN, and the 802.1X guest VLAN on a port. The assignment makes sure the port can correctly process incoming VLAN-tagged traffic.
- On a hybrid port, the guest VLAN can only be an untagged VLAN.
- When you configure multiple security features on a port, follow the guidelines in Table 4.

Table 4 Relationships of the 802.1X guest VLAN and other security features

Feature	Relationship description	Reference
802.1X Auth-Fail VLAN on a port that performs MAC-based access control	The 802.1X Auth-Fail VLAN has a higher priority than the 802.1X guest VLAN.	See "802.1X VLAN manipulation."
Port intrusion protection actions on a port that performs MAC-based access control	The 802.1X guest VLAN feature has higher priority than the block MAC action. The 802.1X guest VLAN feature has lower priority than the shutdown port action of the port intrusion protection feature.	

Prerequisites

Before you configure an 802.1X guest VLAN, complete the following tasks:

- Create the VLAN to be specified as the 802.1X guest VLAN.
- If the 802.1X-enabled port performs MAC-based access control, perform the following operations for the port:
  - Configure the port as a hybrid port.
  - Enable MAC-based VLAN on the port. For more information about MAC-based VLANs, see *Layer 2—LAN Switching Configuration Guide*.
- If the port type is hybrid, verify that the VLAN to be specified as the guest VLAN is not in the tagged VLAN list on the port.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the 802.1X guest VLAN on the port.

   `dot1x guest-vlan` *`guest-vlan-id`*

   By default, no 802.1X guest VLAN exists on a port.

Configuring an 802.1X Auth-Fail VLAN

Restrictions and guidelines

- Assign different IDs to the port VLAN, the voice VLAN, and the 802.1X Auth-Fail VLAN on a port. The assignment makes sure the port can correctly process VLAN-tagged incoming traffic.
- You can configure only one 802.1X Auth-Fail VLAN on a port. The 802.1X Auth-Fail VLANs on different ports can be different.
- On a hybrid port, the Auth-Fail VLAN can only be an untagged VLAN.
- When you configure multiple security features on a port, follow the guidelines in Table 5.

Table 5 Relationships of the 802.1X Auth-Fail VLAN with other features

Feature	Relationship description	Reference
MAC authentication guest VLAN on a port that performs MAC-based access control	The 802.1X Auth-Fail VLAN has a high priority.	See "Configuring MAC authentication."
Port intrusion protection actions on a port that performs MAC-based access control	The 802.1X Auth-Fail VLAN feature has higher priority than the block MAC action. The 802.1X Auth-Fail VLAN feature has lower priority than the shutdown port action of the port intrusion protection feature.	

Prerequisites

Before you configure an 802.1X Auth-Fail VLAN, complete the following tasks:

- Create the VLAN to be specified as the 802.1X Auth-Fail VLAN.
- If the 802.1X-enabled port performs MAC-based access control, perform the following operations for the port:
  - Configure the port as a hybrid port.
  - Enable MAC-based VLAN on the port. For more information about MAC-based VLANs, see *Layer 2—LAN Switching Configuration Guide*.
- If the port type is hybrid, verify that the VLAN to be specified as the Auth-Fail VLAN is not in the tagged VLAN list on the port.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Configure the 802.1X Auth-Fail VLAN on the port.

   ```
   dot1x auth-fail vlan authfail-vlan-id
   ```

   By default, no 802.1X Auth-Fail VLAN exists on a port.

Configuring an 802.1X critical VLAN

Restrictions and guidelines for 802.1X critical VLAN configuration

- Assign different IDs to the PVID, the voice VLAN, and the 802.1X critical VLAN on a port. The assignment makes sure the port can correctly process VLAN-tagged incoming traffic.
- You can configure only one 802.1X critical VLAN on a port. The 802.1X critical VLANs on different ports can be different.
- On a hybrid port, the critical VLAN can only be an untagged VLAN.

Prerequisites

Before you configure an 802.1X critical VLAN, complete the following tasks:

- Create the VLAN to be specified as a critical VLAN.
- If the 802.1X-enabled port performs MAC-based access control, perform the following operations for the port:
  - Configure the port as a hybrid port.
  - Enable MAC-based VLAN on the port. For more information about MAC-based VLANs, see *Layer 2—LAN Switching Configuration Guide*.
- If the port type is hybrid, verify that the VLAN to be specified as the critical VLAN is not in the tagged VLAN list on the port.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the 802.1X critical VLAN on the port.

   dot1x critical vlan *critical-vlan-id*

   By default, no 802.1X critical VLAN exists on a port.

Configuring the authentication trigger feature

About this task

The authentication trigger feature enables the access device to initiate 802.1X authentication when 802.1X clients cannot initiate authentication.

This feature provides the multicast trigger and unicast trigger (see 802.1X authentication initiation in "802.1X overview").

Restrictions and guidelines

- Enable the multicast trigger on a port when the clients attached to the port cannot send EAPOL-Start packets to initiate 802.1X authentication.
- Enable the unicast trigger on a port if only a few 802.1X clients are attached to the port and these clients cannot initiate authentication.
- To avoid duplicate authentication packets, do not enable both triggers on a port.
- As a best practice, do not use the unicast trigger on a port that performs port-based access control. If you do so, users on that port might fail to come online.

Procedure

1. Enter system view.

system-view

2. (Optional.) Set the username request timeout timer.

   dot1x timer tx-period *tx-period-value*

   The default is 30 seconds.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable an authentication trigger.

   dot1x { **multicast-trigger** | **unicast-trigger** }

   By default, the multicast trigger is enabled, and the unicast trigger is disabled.

Setting the maximum number of concurrent 802.1X users on a port

About this task

Perform this task to prevent the system resources from being overused.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the maximum number of concurrent 802.1X users on a port.

   dot1x max-user *max-number*

   The default is 4294967295.

Setting the maximum number of authentication request attempts

About this task

The access device retransmits an authentication request if it does not receive any responses to the request from the client within a period of time. To set the time, use the **dot1x timer tx-period** *tx-period-value* command or the **dot1x timer supp-timeout** *supp-timeout-value* command. The access device stops retransmitting the request if it has made the maximum number of request transmission attempts but still receives no response.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of attempts for sending an authentication request.

   dot1x retry *retries*

   The default setting is 2.

Configuring online user handshake

About this task

The online user handshake feature checks the connectivity status of online 802.1X users. The access device sends handshake requests (EAP-Request/Identity) to online users at the interval specified by the **dot1x timer handshake-period** command. If the device does not receive any EAP-Response/Identity packets from an online user after it has made the maximum handshake attempts, the device sets the user to offline state. To set the maximum handshake attempts, use the **dot1x retry** command.

Typically, the device does not reply to 802.1X clients' EAP-Response/Identity packets with EAP-Success packets. Some 802.1X clients will go offline if they do not receive the EAP-Success packets for handshake. To avoid this issue, enable the online user handshake reply feature.

If iNode clients are deployed, you can also enable the online user handshake security feature to check authentication information in the handshake packets from clients. This feature can prevent 802.1X users that use illegal client software from bypassing iNode security check, such as dual network interface cards (NICs) detection. If a user fails the handshake security checking, the device sets the user to the offline state.

Restrictions and guidelines

- If the network has 802.1X clients that cannot exchange handshake packets with the access device, disable the online user handshake feature. This operation prevents the 802.1X connections from being incorrectly torn down.
- The SmartOn feature and the online user handshake feature are mutually exclusive. Before you enable the online user handshake feature, make sure the SmartOn feature is disabled.
- To use the online user handshake security feature, make sure the online user handshake feature is enabled.
- The online user handshake security feature takes effect only on the network where the iNode client and IMC server are used.
- Enable the online user handshake reply feature only if 802.1X clients will go offline without receiving EAP-Success packets from the device.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the handshake timer.

   dot1x timer handshake-period *handshake-period-value*

   The default is 15 seconds.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable the online user handshake feature.

   dot1x handshake

   By default, the feature is enabled.

5. (Optional.) Enable the online user handshake security feature.

   dot1x handshake secure

   By default, the feature is disabled.

6. (Optional.) Enable the 802.1X online user handshake reply feature.

   dot1x handshake reply enable

By default, the device does not reply to 802.1X clients' EAP-Response/Identity packets during the online handshake process.

Specifying supported domain name delimiters

About this task

By default, the access device supports the at sign (@) as the delimiter. You can also configure the access device to accommodate 802.1X users that use other domain name delimiters. The configurable delimiters include the at sign (@), backslash (\), dot (.), and forward slash (/). Usernames that include domain names can use the format of *username@domain-name, domain-name\username*, *username.domain-name*, or *username/domain-name*.

If an 802.1X username string contains multiple configured delimiters, the rightmost delimiter is the domain name delimiter. For example, if you configure the backslash (\), dot (.), and forward slash (/) as delimiters, the domain name delimiter for the username string 121.123/22\@abc is the backslash (\). The username is **@abc** and the domain name is **121.123/22**.

Restrictions and guidelines

If a username string contains none of the delimiters, the access device authenticates the user in the mandatory or default ISP domain.

If you configure the access device to send usernames with domain names to the RADIUS server, make sure the domain delimiter can be recognized by the RADIUS server. For username format configuration, see the **`user-name-format`** command in *Security Command Reference*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify a set of domain name delimiters for 802.1X users.

   ```
   dot1x domain-delimiter string
   ```

   By default, only the at sign (@) delimiter is supported.

Configuring the EAD assistant feature

Restrictions and guidelines

- To make the EAD assistant feature take effect on an 802.1X-enabled port, you must set the port authorization mode to **auto**.
- For the 802.1X guest VLAN feature to work correctly, do not enable EAD assistant together with the 802.1X guest VLAN feature.
- If you use free IP and Auth-Fail VLAN features together, make sure the resources in the Auth-Fail VLAN are on the free IP segments.
- To allow a user to obtain a dynamic IP address before it passes 802.1X authentication, make sure the DHCP server is on the free IP segment.
- The server that provides the redirect URL must be on the free IP accessible to unauthenticated users.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the EAD assistant feature.

   ```
   dot1x ead-assistant enable
   ```

By default, this feature is disabled.

3. Configure a free IP.

   dot1x ead-assistant free-ip *ip-address* { *mask-length* | *mask-address* }

   By default, no free IPs exist.

   Repeat this command to configure multiple free IPs.

4. (Optional.) Configure the redirect URL if users will use Web browsers to access the network.

   dot1x ead-assistant url *url-string*

   By default, no redirect URL exists.

5. (Optional.) Set the EAD rule timer.

   dot1x timer ead-timeout *ead-timeout-value*

   The default setting is 30 minutes.

   To avoid using up ACL resources when a large number of EAD users exist, you can shorten the EAD rule timer.

Configuring 802.1X SmartOn

About this task

When the device sends a unicast EAP-Request/Notification packet to the client, it starts the SmartOn client timeout timer (set by using the **dot1x smarton timer supp-timeout** command).

- If the device does not receive any EAP-Response/Notification packets from the client within the timeout timer, it retransmits the EAP-Request/Notification packet to the client. After the device has made the maximum retransmission attempts but received no response, it stops the 802.1X authentication process for the client.
- If the device receives an EAP-Response/Notification packet within the timer or before the maximum retransmission attempts have been made, it starts the SmartOn authentication. If the SmartOn switch ID and the MD5 digest of the SmartOn password in the packet match those on the device, 802.1X authentication continues for the client. Otherwise, the device denies the client's 802.1X authentication request.

Restrictions and guidelines

The SmartOn feature is mutually exclusive with the 802.1X online user handshake feature.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the SmartOn feature on the port.

   dot1x smarton

   By default, this feature is disabled.

4. Return to system view.

   quit

5. Configure the SmartOn switch ID.

   dot1x smarton switchid *switch-string*

   By default, no SmartOn switch ID exists.

6. Set the SmartOn password.

   dot1x smarton password { **cipher** | **simple** } *string*

By default, no SmartOn password exists.

7. (Optional.) Set the SmartOn client timeout timer.

   dot1x smarton timer supp-timeout *supp-timeout-value*

   The default timer is 30 seconds.

8. (Optional.) Set the maximum attempts for retransmitting an EAP-Request/Notification packet to a client.

   dot1x smarton retry *retries*

   By default, the device allows a maximum of 3 attempts for retransmitting an EAP-Request/Notification packet to a client.

Display and maintenance commands for 802.1X

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display 802.1X session information, statistics, or configuration information of specified or all ports.	**display dot1x** [**sessions** | **statistics**] [**interface** *interface-type interface-number*]
Display online 802.1X user information.	**display dot1x connection** [**interface** *interface-type interface-number* | **slot** *slot-number* | **user-mac** *mac-address* | **user-name** *name-string*]
Remove users from the 802.1X guest VLAN on a port.	**reset dot1x guest-vlan interface** *interface-type interface-number* [**mac-address** *mac-address*]
Clear 802.1X statistics.	**reset dot1x statistics** [**interface** *interface-type interface-number*]

802.1X authentication configuration examples

Example: Configuring basic 802.1X authentication

Network configuration

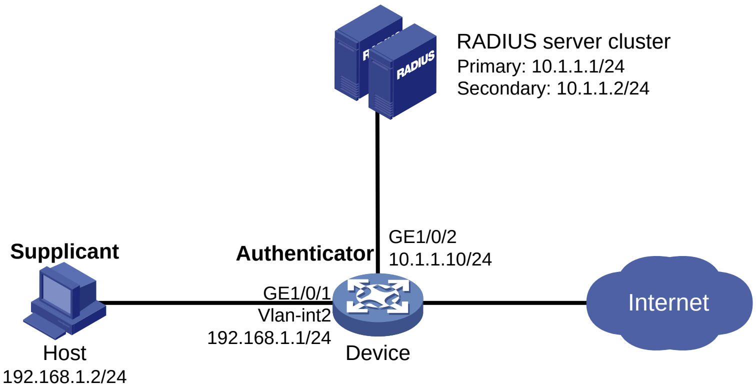
As shown in Figure 12, the access device performs 802.1X authentication for users that connect to GigabitEthernet 1/0/1. Implement MAC-based access control on the port, so the logoff of one user does not affect other online 802.1X users.

Use RADIUS servers to perform authentication, authorization, and accounting for the 802.1X users. If RADIUS authentication fails, perform local authentication on the access device.

Configure the RADIUS server at 10.1.1.1/24 as the primary authentication and accounting server, and the RADIUS server at 10.1.1.2/24 as the secondary authentication and accounting server. Assign all users to ISP domain **bbb**.

Set the shared key to **name** for packets between the access device and the authentication server. Set the shared key to **money** for packets between the access device and the accounting server.

Figure 12 Network diagram



Procedure

For information about the RADIUS commands used on the access device in this example, see *Security Command Reference*.

1. Configure the RADIUS servers and add user accounts for the 802.1X users. Make sure the RADIUS servers can provide authentication, authorization, and accounting services. (Details not shown.)
2. Configure IP addresses for interfaces, as shown in Figure 12. (Details not shown.)
3. Configure user accounts for the 802.1X users on the access device:

   # Add a local network access user with username **localuser** and password **123456TESTplat&!** in plaintext. (Make sure the username and password are the same as those configured on the RADIUS servers.)

   ```
   <Device> system-view
   [Device] local-user localuser class network
   [Device-luser-network-localuser] password simple 123456TESTplat&!
   ```

   # Set the service type to **lan-access**.

   ```
   [Device-luser-network-localuser] service-type lan-access
   [Device-luser-network-localuser] quit
   ```

4. Configure a RADIUS scheme on the access device:

   # Create a RADIUS scheme named **radius1** and enter RADIUS scheme view.

   ```
   [Device] radius scheme radius1
   ```

   # Specify the IP addresses of the primary authentication and accounting RADIUS servers.

   ```
   [Device-radius-radius1] primary authentication 10.1.1.1
   [Device-radius-radius1] primary accounting 10.1.1.1
   ```

   # Configure the IP addresses of the secondary authentication and accounting RADIUS servers.

   ```
   [Device-radius-radius1] secondary authentication 10.1.1.2
   [Device-radius-radius1] secondary accounting 10.1.1.2
   ```

   # Specify the shared key between the access device and the authentication server.

   ```
   [Device-radius-radius1] key authentication simple name
   ```

# Specify the shared key between the access device and the accounting server.

```
[Device-radius-radius1] key accounting simple money
```

# Exclude the ISP domain names from the usernames sent to the RADIUS servers.

```
[Device-radius-radius1] user-name-format without-domain
[Device-radius-radius1] quit
```

NOTE:

The access device must use the same username format as the RADIUS server. If the RADIUS server includes the ISP domain name in the username, so must the access device.

5. Configure the ISP domain on the access device:

   # Create an ISP domain named **bbb** and enter ISP domain view.

   ```
   [Device] domain bbb
   ```

   # Apply RADIUS scheme **radius1** to the ISP domain, and specify local authentication as the secondary authentication method.

   ```
   [Device-isp-bbb] authentication lan-access radius-scheme radius1 local
   [Device-isp-bbb] authorization lan-access radius-scheme radius1 local
   [Device-isp-bbb] accounting lan-access radius-scheme radius1 local
   [Device-isp-bbb] quit
   ```

6. Configure 802.1X on the access device:

   # Enable 802.1X on GigabitEthernet 1/0/1.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] dot1x
   ```

   # Enable MAC-based access control on the port. By default, the port uses MAC-based access control.

   ```
   [Device-GigabitEthernet1/0/1] dot1x port-method macbased
   ```

   # Specify ISP domain **bbb** as the mandatory domain.

   ```
   [Device-GigabitEthernet1/0/1] dot1x mandatory-domain bbb
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Enable 802.1X globally.

   ```
   [Device] dot1x
   ```

7. Configure the 802.1X client. If an iNode client is used, do not select the **Carry version info** option in the client configuration. (Details not shown.)

Verifying the configuration

# Verify the 802.1X configuration on GigabitEthernet 1/0/1.

```
[Device] display dot1x interface gigabitethernet 1/0/1
```

# Display the user connection information after an 802.1X user passes authentication.

```
[Device] display dot1x connection
```

Example: Configuring 802.1X guest VLAN and authorization VLAN

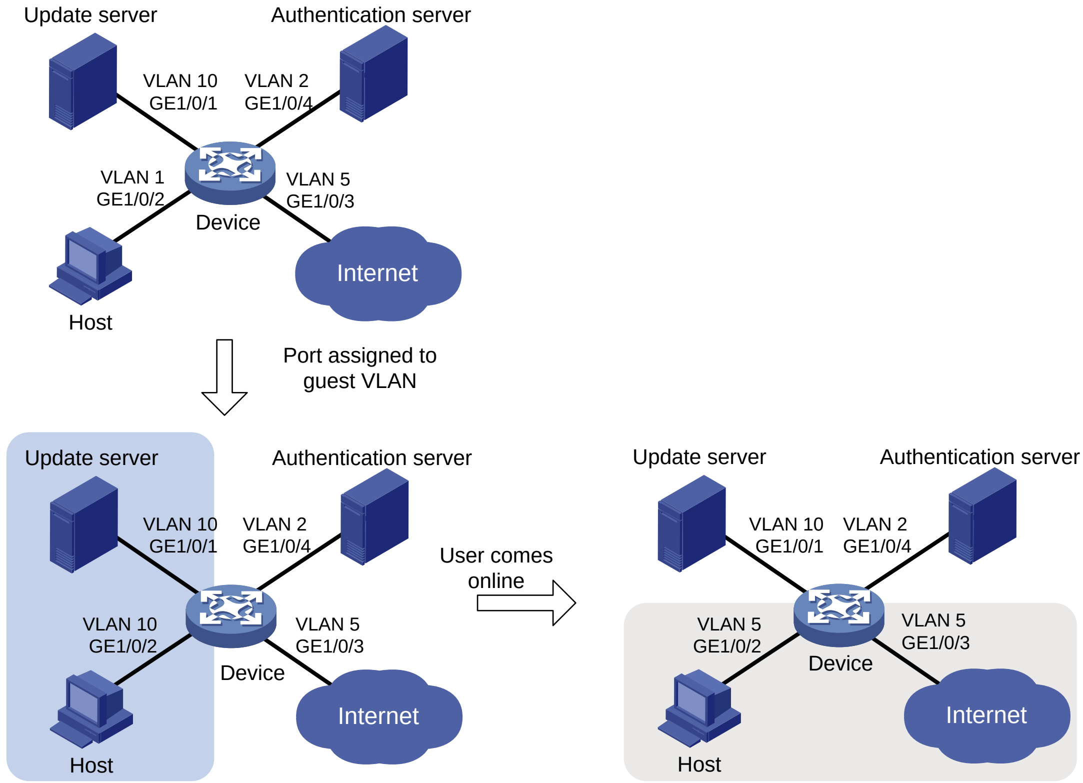
Network configuration

As shown in Figure 13:

- Use RADIUS servers to perform authentication, authorization, and accounting for 802.1X users that connect to GigabitEthernet 1/0/2. Implement port-based access control on the port.

- Configure VLAN 10 as the 802.1X guest VLAN on GigabitEthernet 1/0/2. The host and the update server are both in VLAN 10, and the host can access the update server and download the 802.1X client software.
- Configure a QoS policy to deny packets destined for the Internet (5.1.1.1) and apply the QoS policy to the outbound direction of VLAN 10. The configuration prevents users in the 802.1X guest VLAN from accessing the Internet before they pass 802.1X authentication.
- After the host passes 802.1X authentication, the access device assigns the host to VLAN 5 where GigabitEthernet 1/0/3 is. The host can access the Internet.

Figure 13 Network diagram



Procedure

For information about the RADIUS commands used on the access device in this example, see *Security Command Reference*.

1. Configure the RADIUS server to provide authentication, authorization, and accounting services. Configure user accounts and authorization VLAN (VLAN 5 in this example) for the users. (Details not shown.)
2. Create VLANs, and assign ports to the VLANs on the access device.

NOTE:

By default, VLAN 1 exists and all ports belong to the VLAN. You do not need to create the VLAN or assign GigabitEthernet 1/0/2 to the VLAN.

```
<Device> system-view
[Device] vlan 10
[Device-vlan10] port gigabitethernet 1/0/1
[Device-vlan10] quit
[Device] vlan 2
[Device-vlan2] port gigabitethernet 1/0/4
```

```
[Device-vlan2] quit
[Device] vlan 5
[Device-vlan5] port gigabitethernet 1/0/3
[Device-vlan5] quit
```

3. Configure a QoS policy:

   # Configure advanced ACL 3000 to match packets destined for 5.1.1.1.

   ```
   [Device] acl advanced 3000
   [Device-acl-ipv4-adv-3000] rule permit ip destination 5.1.1.1 0.0.0.255
   [Device-acl-ipv4-adv-3000] quit
   ```

   # Specify advanced ACL 3000 in traffic class **classifier_1** to match traffic.

   ```
   [Device] traffic classifier classifier_1
   [Device-classifier-classifier_1] if-match acl 3000
   [Device-classifier-classifier_1] quit
   ```

   # Configure traffic behavior **behavior_1** to deny matching packets.

   ```
   [Device] traffic behavior behavior_1
   [Device-behavior-behavior_1] filter deny
   [Device-behavior-behavior_1] quit
   ```

   # Configure QoS policy **policy_1** to associate traffic class **classifier_1** with traffic behavior **behavior_1**.

   ```
   [Device] qos policy policy_1
   [Device-qospolicy-policy_1] classifier classifier_1 behavior behavior_1
   [Device-qospolicy-policy_1] quit
   ```

   # Apply QoS policy **policy_1** to the outbound direction of VLAN 10.

   ```
   [Device] qos vlan-policy policy_1 vlan 10 outbound
   ```

4. Configure a RADIUS scheme on the access device:

   # Create RADIUS scheme **2000** and enter RADIUS scheme view.

   ```
   [Device] radius scheme 2000
   ```

   # Specify the server at 10.11.1.1 as the primary authentication server, and set the authentication port to 1812.

   ```
   [Device-radius-2000] primary authentication 10.11.1.1 1812
   ```

   # Specify the server at 10.11.1.1 as the primary accounting server, and set the accounting port to 1813.

   ```
   [Device-radius-2000] primary accounting 10.11.1.1 1813
   ```

   # Set the shared key to **abc** in plain text for secure communication between the authentication server and the device.

   ```
   [Device-radius-2000] key authentication simple abc
   ```

   # Set the shared key to **abc** in plain text for secure communication between the accounting server and the device.

   ```
   [Device-radius-2000] key accounting simple abc
   ```

   # Exclude the ISP domain names from the usernames sent to the RADIUS server.

   ```
   [Device-radius-2000] user-name-format without-domain
   [Device-radius-2000] quit
   ```

5. Configure an ISP domain on the access device:

   # Create ISP domain **bbb** and enter ISP domain view.

   ```
   [Device] domain bbb
   ```

   # Apply RADIUS scheme 2000 to the ISP domain for authentication, authorization, and accounting.

   ```
   [Device-isp-bbb] authentication lan-access radius-scheme 2000
   ```

```
[Device-isp-bbb] authorization lan-access radius-scheme 2000
[Device-isp-bbb] accounting lan-access radius-scheme 2000
[Device-isp-bbb] quit
```

6. Configure 802.1X on the access device:

   # Enable 802.1X on GigabitEthernet 1/0/2.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] dot1x
   ```

   # Implement port-based access control on the port.

   ```
   [Device-GigabitEthernet1/0/2] dot1x port-method portbased
   ```

   # Set the port authorization mode to **auto**. By default, the port uses the auto mode.

   ```
   [Device-GigabitEthernet1/0/2] dot1x port-control auto
   ```

   # Specify VLAN 10 as the 802.1X guest VLAN on GigabitEthernet 1/0/2.

   ```
   [Device-GigabitEthernet1/0/2] dot1x guest-vlan 10
   [Device-GigabitEthernet1/0/2] quit
   ```

   # Enable 802.1X globally.

   ```
   [Device] dot1x
   ```

7. Configure the 802.1X client. Make sure the 802.1X client can update its IP address after the access port is assigned to the guest VLAN or an authorization VLAN. (Details not shown.)

Verifying the configuration

# Verify the 802.1X guest VLAN configuration on GigabitEthernet 1/0/2.

```
[Device] display dot1x interface gigabitethernet 1/0/2
```

# Verify that GigabitEthernet 1/0/2 is assigned to VLAN 10 before any user passes authentication on the port.

```
[Device] display vlan 10
```

# After a user passes authentication, display information on GigabitEthernet 1/0/2. Verify that GigabitEthernet 1/0/2 is assigned to VLAN 5.

```
[Device] display interface gigabitethernet 1/0/2
```

Example: Configuring 802.1X with ACL assignment

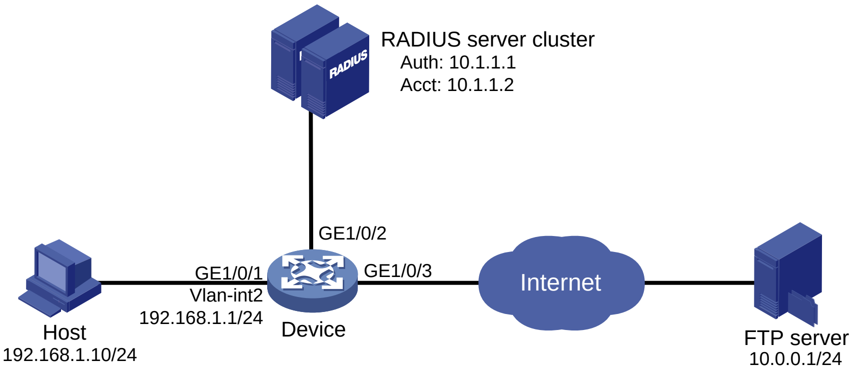
Network configuration

As shown in Figure 14, the host that connects to GigabitEthernet 1/0/1 must pass 802.1X authentication to access the Internet.

Perform 802.1X authentication on GigabitEthernet 1/0/1. Use the RADIUS server at 10.1.1.1 as the authentication and authorization server, and the RADIUS server at 10.1.1.2 as the accounting server.

Configure ACL assignment on GigabitEthernet 1/0/1 to deny access of 802.1X users to the FTP server from 8:00 to 18:00 on weekdays.

Figure 14 Network diagram



Procedure

For information about the RADIUS commands used on the access device in this example, see *Security Command Reference*.

1. Configure the RADIUS servers to provide authentication, authorization, and accounting services. Add user accounts and specify the ACL (ACL 3000 in this example) for the users. (Details not shown.)
2. Configure IP addresses for interfaces, as shown in Figure 14. (Details not shown.)
3. Configure a RADIUS scheme on the access device:

# Create RADIUS scheme **2000** and enter RADIUS scheme view.

```
<Device> system-view
[Device] radius scheme 2000
```

# Specify the server at 10.1.1.1 as the primary authentication server, and set the authentication port to 1812.

```
[Device-radius-2000] primary authentication 10.1.1.1 1812
```

# Specify the server at 10.1.1.2 as the primary accounting server, and set the accounting port to 1813.

```
[Device-radius-2000] primary accounting 10.1.1.2 1813
```

# Set the shared key to **abc** in plain text for secure communication between the authentication server and the device.

```
[Device-radius-2000] key authentication simple abc
```

# Set the shared key to **abc** in plain text for secure communication between the accounting server and the device.

```
[Device-radius-2000] key accounting simple abc
```

# Exclude the ISP domain names from the usernames sent to the RADIUS server.

```
[Device-radius-2000] user-name-format without-domain
[Device-radius-2000] quit
```

4. Configure an ISP domain on the access device:

# Create ISP domain **bbb** and enter ISP domain view.

```
[Device] domain bbb
```

# Apply RADIUS scheme 2000 to the ISP domain for authentication, authorization, and accounting.

```
[Device-isp-bbb] authentication lan-access radius-scheme 2000
[Device-isp-bbb] authorization lan-access radius-scheme 2000
[Device-isp-bbb] accounting lan-access radius-scheme 2000
[Device-isp-bbb] quit
```

5. Configure a time range named **ftp** from 8:00 to 18:00 on weekdays on the access device.

   ```
   [Device] time-range ftp 8:00 to 18:00 working-day
   ```

6. Configure ACL 3000 to deny packets destined for the FTP server at 10.0.0.1 during the specified time range on the access device.

   ```
   [Device] acl advanced 3000
   [Device-acl-ipv4-adv-3000] rule 0 deny ip destination 10.0.0.1 0 time-range ftp
   [Device-acl-ipv4-adv-3000] quit
   ```

7. Configure 802.1X on the access device:

   # Enable 802.1X on GigabitEthernet 1/0/1.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] dot1x
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Enable 802.1X globally.

   ```
   [Device] dot1x
   ```

8. Configure the 802.1X client. Make sure the client is able to update its IP address after the access port is assigned to the 802.1X guest VLAN or an authorization VLAN. (Details not shown.)

Verifying the configuration

# Use the user account to pass authentication. (Details not shown.)

# Verify that the user cannot ping the FTP server at any time from 8:00 to 18:00 on any weekday.

```
C:\>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

The output shows that ACL 3000 is active on the user, and the user cannot access the FTP server.

Example: Configuring 802.1X with EAD assistant (with DHCP relay agent)

Network configuration

As shown in Figure 15:

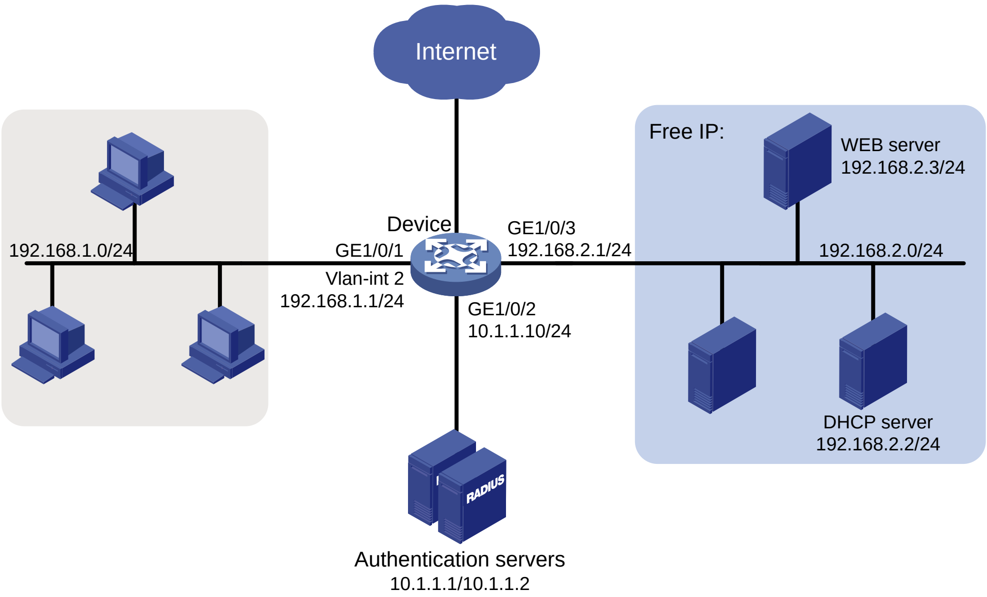
- The intranet 192.168.1.0/24 is attached to GigabitEthernet 1/0/1 of the access device.
- The hosts use DHCP to obtain IP addresses.
- A DHCP server and a Web server are deployed on the 192.168.2.0/24 subnet for users to obtain IP addresses and download client software.

Deploy an EAD solution for the intranet to meet the following requirements:

- Allow unauthenticated users and users that have failed 802.1X authentication to access 192.168.2.0/24. The users can obtain IP addresses and download software.

- If these users use a Web browser to access a network other than 192.168.2.0/24, redirect them to the Web server for 802.1X client downloading.
- Allow authenticated 802.1X users to access the network.

Figure 15 Network diagram



Procedure

1. Make sure the DHCP server, the Web server, and the authentication servers have been configured correctly. (Details not shown.)
2. Configure IP addresses for interfaces, as shown in Figure 15. (Details not shown.)
3. Configure DHCP relay:

   # Enable DHCP.

   ```
   <Device> system-view
   [Device] dhcp enable
   ```

   # Enable the DHCP relay agent on VLAN-interface 2.

   ```
   [Device] interface vlan-interface 2
   [Device-Vlan-interface2] dhcp select relay
   ```

   # Specify the DHCP server 192.168.2.2 on the relay agent interface VLAN-interface 2.

   ```
   [Device-Vlan-interface2] dhcp relay server-address 192.168.2.2
   [Device-Vlan-interface2] quit
   ```

4. Configure a RADIUS scheme:

   # Create RADIUS scheme **2000** and enter RADIUS scheme view.

   ```
   [Device] radius scheme 2000
   ```

   # Specify the server at 10.1.1.1 as the primary authentication server, and set the authentication port to 1812.

   ```
   [Device-radius-2000] primary authentication 10.1.1.1 1812
   ```

   # Specify the server at 10.1.1.2 as the primary accounting server, and set the accounting port to 1813.

   ```
   [Device-radius-2000] primary accounting 10.1.1.2 1813
   ```

   # Set the shared key to **abc** in plain text for secure communication between the authentication server and the device.

   ```
   [Device-radius-2000] key authentication simple abc
   ```

# Set the shared key to **abc** in plain text for secure communication between the accounting server and the device.

```
[Device-radius-2000] key accounting simple abc
```

# Exclude the ISP domain names from the usernames sent to the RADIUS server.

```
[Device-radius-2000] user-name-format without-domain
[Device-radius-2000] quit
```

5. Configure an ISP domain:

   # Create ISP domain **bbb** and enter ISP domain view.

   ```
   [Device] domain bbb
   ```

   # Apply RADIUS scheme 2000 to the ISP domain for authentication, authorization, and accounting.

   ```
   [Device-isp-bbb] authentication lan-access radius-scheme 2000
   [Device-isp-bbb] authorization lan-access radius-scheme 2000
   [Device-isp-bbb] accounting lan-access radius-scheme 2000
   [Device-isp-bbb] quit
   ```

6. Configure 802.1X:

   # Configure the free IP.

   ```
   [Device] dot1x ead-assistant free-ip 192.168.2.0 24
   ```

   # Configure the redirect URL for client software download.

   ```
   [Device] dot1x ead-assistant url http://192.168.2.3
   ```

   # Enable the EAD assistant feature.

   ```
   [Device] dot1x ead-assistant enable
   ```

   # Enable 802.1X on GigabitEthernet 1/0/1.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] dot1x
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Enable 802.1X globally.

   ```
   [Device] dot1x
   ```

Verifying the configuration

# Verify the 802.1X configuration.

```
[Device] display dot1x
```

# Verify that you can ping an IP address on the free IP subnet from a host. (Details not shown.)

# Verify that you are redirected to the Web server when you enter in your Web browser an IP address not on the free IP. (Details not shown.)

Example: Configuring 802.1X with EAD assistant (with DHCP server)

Network configuration

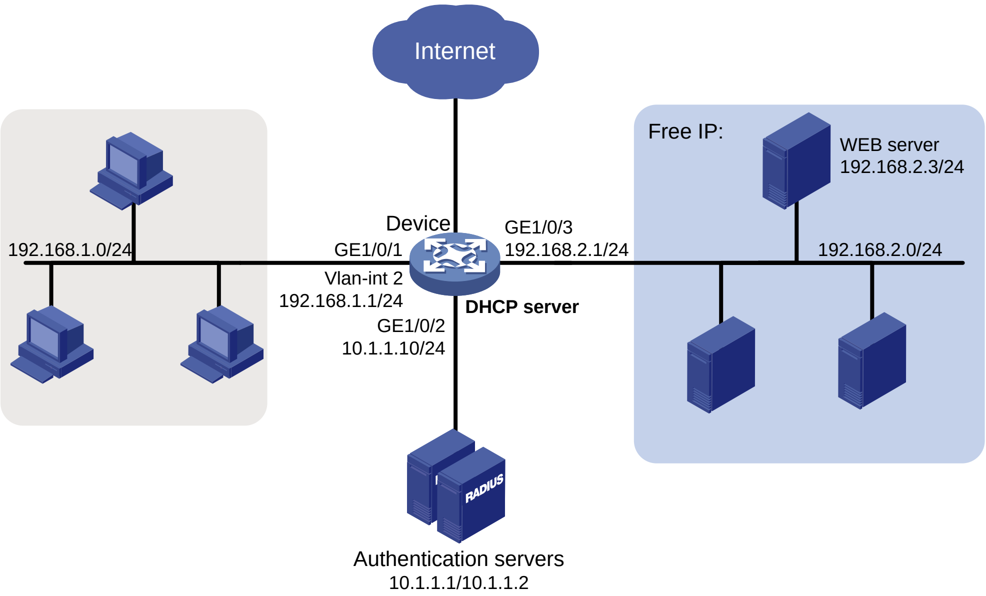
As shown in Figure 16:

- The intranet 192.168.1.0/24 is attached to GigabitEthernet 1/0/1 of the access device.
- The hosts use DHCP to obtain IP addresses.
- A Web server is deployed on the 192.168.2.0/24 subnet for users to download client software.

Deploy an EAD solution for the intranet to meet the following requirements:

- Allow unauthenticated users and users that have failed 802.1X authentication to access 192.168.2.0/24. The users can download software.
- If these users use a Web browser to access a network other than 192.168.2.0/24, redirect them to the Web server for 802.1X client downloading.
- Allow authenticated 802.1X users to access the network.

Figure 16 Network diagram



Procedure

1. Make sure the Web server and the authentication servers have been configured correctly. (Details not shown.)
2. Configure IP addresses for interfaces, as shown in Figure 16. (Details not shown.)
3. Configure the DHCP server:

   # Enable DHCP.

```
<Device> system-view
[Device] dhcp enable
```

   # Enable the DHCP server on VLAN-interface 2.

```
[Device] interface vlan-interface 2
[Device-Vlan-interface2] dhcp select server
[Device-Vlan-interface2] quit
```

   # Create DHCP address pool **0**.

```
[Device] dhcp server ip-pool 0
```

   # Specify subnet 192.168.1.0/24 in DHCP address pool 0.

```
[Device-dhcp-pool-0] network 192.168.1.0 mask 255.255.255.0
```

   # Specify the gateway address 192.168.1.1 in DHCP address pool 0.

```
[Device-dhcp-pool-0] gateway-list 192.168.1.1
[Device-dhcp-pool-0] quit
```

4. Configure a RADIUS scheme:

   # Create RADIUS scheme **2000** and enter RADIUS scheme view.

```
[Device] radius scheme 2000
```

# Specify the server at 10.1.1.1 as the primary authentication server, and set the authentication port to 1812.

```
[Device-radius-2000] primary authentication 10.1.1.1 1812
```

# Specify the server at 10.1.1.2 as the primary accounting server, and set the accounting port to 1813.

```
[Device-radius-2000] primary accounting 10.1.1.2 1813
```

# Set the shared key to **abc** in plain text for secure communication between the authentication server and the device.

```
[Device-radius-2000] key authentication simple abc
```

# Set the shared key to **abc** in plain text for secure communication between the accounting server and the device.

```
[Device-radius-2000] key accounting simple abc
```

# Exclude the ISP domain names from the usernames sent to the RADIUS server.

```
[Device-radius-2000] user-name-format without-domain
[Device-radius-2000] quit
```

5. Configure an ISP domain:

# Create ISP domain **bbb** and enter ISP domain view.

```
[Device] domain bbb
```

# Apply RADIUS scheme 2000 to the ISP domain for authentication, authorization, and accounting.

```
[Device-isp-bbb] authentication lan-access radius-scheme 2000
[Device-isp-bbb] authorization lan-access radius-scheme 2000
[Device-isp-bbb] accounting lan-access radius-scheme 2000
[Device-isp-bbb] quit
```

6. Configure 802.1X:

# Configure the free IP.

```
[Device] dot1x ead-assistant free-ip 192.168.2.0 24
```

# Configure the redirect URL for client software download.

```
[Device] dot1x ead-assistant url http://192.168.2.3
```

# Enable the EAD assistant feature.

```
[Device] dot1x ead-assistant enable
```

# Enable 802.1X on GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] dot1x
[Device-GigabitEthernet1/0/1] quit
```

# Enable 802.1X globally.

```
[Device] dot1x
```

Verifying the configuration

# Verify the 802.1X configuration.

```
[Device] display dot1x
```

# Verify that you can ping an IP address on the free IP subnet from a host. (Details not shown.)

# Verify that you are redirected to the Web server when you enter in your Web browser an IP address not on the free IP. (Details not shown.)

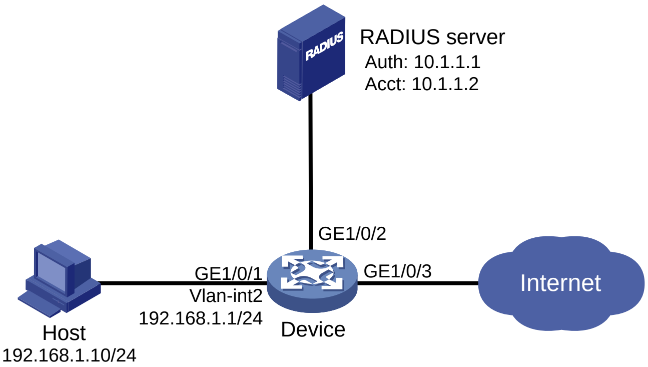
Example: Configuring 802.1X SmartOn

Network configuration

As shown in Figure 17, configure the SmartOn feature on GigabitEthernet 1/0/1 so that the host must pass SmartOn authentication before 802.1X authentication.

Set the SmartOn password to **1234** in plain text and switch ID to **XYZ**. Set the SmartOn client timeout timer to 40 seconds.

Figure 17 Network diagram



Procedure

1. Configure a RADIUS scheme:

   # Create RADIUS scheme **2000** and enter RADIUS scheme view.

   ```
   <Device> system-view
   [Device] radius scheme 2000
   ```

   # Specify the server at 10.1.1.1 as the primary authentication server, and set the authentication port to 1812.

   ```
   [Device-radius-2000] primary authentication 10.1.1.1 1812
   ```

   # Specify the server at 10.1.1.2 as the primary accounting server, and set the accounting port to 1813.

   ```
   [Device-radius-2000] primary accounting 10.1.1.2 1813
   ```

   # Set the shared key to **abc** in plain text for secure communication between the authentication server and the device.

   ```
   [Device-radius-2000] key authentication simple abc
   ```

   # Set the shared key to **abc** in plain text for secure communication between the accounting server and the device.

   ```
   [Device-radius-2000] key accounting simple abc
   ```

   # Exclude the ISP domain names from the usernames sent to the RADIUS server.

   ```
   [Device-radius-2000] user-name-format without-domain
   [Device-radius-2000] quit
   ```

2. Configure an ISP domain:

   # Create ISP domain **bbb** and enter ISP domain view.

   ```
   [Device] domain bbb
   ```

   # Apply RADIUS scheme 2000 to the ISP domain for authentication, authorization, and accounting.

   ```
   [Device-isp-bbb] authentication lan-access radius-scheme 2000
   [Device-isp-bbb] authorization lan-access radius-scheme 2000
   ```

```
[Device-isp-bbb] accounting lan-access radius-scheme 2000
[Device-isp-bbb] quit
```

3. Configure 802.1X and SmartOn:

   # Enable 802.1X on GigabitEthernet 1/0/1.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] dot1x
   ```

   # Enable SmartOn on GigabitEthernet 1/0/1.

   ```
   [Device-GigabitEthernet1/0/1] dot1x smarton
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Set the SmartOn password to **1234** in plain text and the switch ID to **XYZ**.

   ```
   [Device] dot1x smarton password simple 1234
   [Device] dot1x smarton switchid XYZ
   ```

   # Set the SmartOn client timeout timer to 40 seconds.

   ```
   [Device] smarton timer supp-timeout 40
   ```

   # Enable 802.1X globally.

   ```
   [Device] dot1x
   ```

Troubleshooting 802.1X

EAD assistant URL redirection failure

Symptom

Unauthenticated users are not redirected to the specified redirect URL after they enter external website addresses in their Web browsers.

Analysis

Redirection will not happen for one of the following reasons:

- The address is in the string format. The operating system of the host regards the string as a website name and tries to resolve the string. If the resolution fails, the operating system sends an ARP request, but the target address is not in the dotted decimal notation. The redirection feature does redirect this kind of ARP request.
- The address is within a free IP segment. No redirection will take place, even if no host is present with the address.
- The redirect URL is not in a free IP segment.
- No server is using the redirect URL, or the server with the URL does not provide Web services.

Solution

To resolve the issue:

1. Enter a dotted decimal IP address that is not in any free IP segments.
2. Verify that the access device and the server are configured correctly.
3. If the issue persists, contact NSFOCUS Support.

Contents

Configuring user identification

About user identification

The user identification feature works with other security features such as blacklist and object policy to control users' network access based on the users' IP addresses or MAC addresses. This feature provides facilitation for network administrators to deploy policies for security features and to collect statistics and analysis for network attack behaviors and traffic flow on a per-user basis. It also enables the network administrators to implement policy control regardless of changes to the user IP or MAC addresses.

Identity-based user access control

The following shows the process for identity-based user access control:

1. Identity authentication. A network access user passes identity authentication and comes online.
2. User identification. The device obtains the username and IP or MAC address of the online user, and associates the information with the local identity user account and the local identity group. Then, the username-IP or username-MAC mapping for the network access user is created. The administrator can also add static username-IP or username-MAC mappings to permit network access without identity authentication.
3. Identity-based access control. The device identifies the source IP or MAC address of the traffic destined for the network, and resolves the IP or MAC address to the username and user group based on the mapping. The device performs network access control for the user or user group based on other security feature settings such as blacklist and object policy.

Identity user management

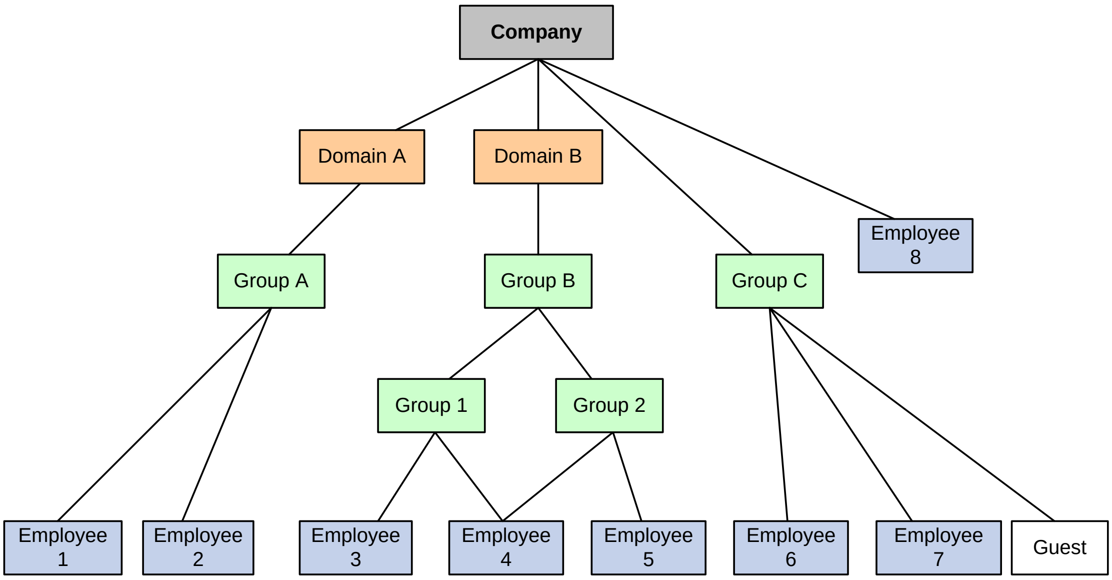
All identity users are organized in a tree structure. Identity user, identity group, and identity domain are management units, and their hierarchy levels are in ascending order.

- An identity user can belong to one or multiple identity groups.
- An identity group can belong to one or multiple higher-layer identity groups.
- An identity user or an identity group can belong to no or one identity domain.

The device uniquely identifies a managed object by the combination of identity domain and username or the combination of identity domain and identity group.

The tree structure facilitates user location and query. As shown in Figure 1, the network administrator creates identity groups and identity users for different departments and employees to implement identity-based network access control.

Figure 1 Identity user management architecture



Identity user accounts

Identity user accounts are used to record identification information of network access users from different sources. The identification information includes the username, user group name, and identity domain name of the users. The user identification module uniformly manages identity users from different sources.

The device supports the following methods to create identity user accounts:

- **Learning from the local user database**—The user identification module learns network access user information from the local user database and saves the user information as identity user accounts. For more information about network access users, see "Configuring AAA."
- **Importing from a .csv file**—The network administrator imports user information from a .csv file to the device and the device automatically creates identity user accounts based on the imported information.
- **Importing from remote servers**—The device initiates user information requests to remote servers, imports network access user information, and then creates identity user accounts based on the imported information. This method enables the network administrator to manage identity user accounts when user information is on the remote servers. Supported remote servers include LDAP servers and IMC RESTful servers.

Online identity users

Online identity users are online network access users that are managed by the user identification module. The device records the username, identity domain name, IP address, and MAC address of online identity users.

Online identity users include dynamic online identity users and static online identity users.

Dynamic online identity users

Dynamic online identity users have the following sources:

- **Online network access users that access the network through the device**—After a user passes local or remote authentication and comes online, the user identification module searches the user's username and domain name in local identity user accounts. If a matching entry is found, the device creates an online identity user entry for the user.

- **Online network access users obtained from remote servers**—After the device obtains information about an online user from a remote server, the user identification module searches the user's username and domain name in local identity user accounts. If a matching entry is found, the device creates an online identity user entry for the user. The device can obtain information about all online users of remote servers (including online users on the other devices) for unified management and monitoring.

  Supported remote servers include security management servers and IMC RESTful servers.

  - The device obtains online network access user information pushed by security management servers.
  - The device actively imports online network access user information from an IMC RESTful server.

Static online identity users

Static online identity users originate from static identity users configured by the network administrator. Each static identity user contains the mapping between the username and the IP or MAC addresses of the user. After a static identity user is created, the user identification module searches the user's username and domain name in local identity user accounts. If a matching entry is found, the device creates a static online identity user entry for the static identity user. Static online identity users can access the network without identity authentication but their access to the network is controlled by security features. The network administrator can configure static identity users when only few people need to temporarily access the network.

Application of online identity users

Application modules can impose policies of security features on online identity users. When online identity user entries are deleted, the user identification module will instruct the application modules to stop processing services for the users.

Identity groups

Identity users can be added to different groups for batch configuration and hierarchical user management. The groups are called identity groups. The user identification module uniformly manages identity groups from different sources.

Creation of identity groups

The device supports the following methods to create identity groups:

- **Learning from the local user database**—When a local user group is created, the device instructs the user identification module to create an identity group with the same group name. For more information about local user groups, see "Configuring AAA."
- **Importing from a .csv file**—The device imports identity user account information from a .csv file and then automatically creates identity groups based on the imported information.
- **Importing from remote servers**—The device can import identity user account information from an IMC RESTful server or LDAP servers and then create identity groups based on the group information in the accounts. The device can also directly obtain user group information from LDAP servers and then creates identity groups.

Application of identity groups

An identity group is activated when it is used by an application module, and all services based on the identity group will take effect. When the application module stops using the identity group, the identity group is inactive.

Restrictions and guidelines: User identification configuration

The user identity feature on the device can interact with the following remote servers to learn user information:

- **IMC RESTful server**—The device imports identity user accounts, identity groups, and online identity users from the IMC RESTful server.
- **LDAP servers**—The device imports identity user accounts and identity groups from LDAP servers.
- **Security management servers**—The device obtains online identity users from security management servers.

To use the IMC RESTful server, make sure the server runs IMC PLAT 7.3 (E0605P04) installed with the SSM E0503P04 component or IMC PLAT 7.3(E0605) installed with the EIA E0512 component.

User identification is not applicable to portal users that perform MAC-based quick portal authentication. For more information about MAC-based quick portal authentication, see "Configuring portal authentication."

User identification tasks at a glance

To configure user identification, perform the following tasks:

1. Enabling the user identification feature
2. Configuring remote servers and an identity user import policy

   Perform this task if the device needs to import user information from RESTful or LDAP servers.

   a. Configure remote server parameters

      Configuring a RESTful server

      Configuring an LDAP scheme

   b. Configuring an identity user import policy
3. Configuring a security management server set

   Perform this task if the device needs to obtain user information from security management servers.
4. (Optional.) Managing identity user accounts
   - Enabling automatic identity user account import
   - Manually importing identity user accounts from remote servers
   - Manually importing identity user accounts from a .csv file
   - Manually exporting identity user accounts
   - Deleting identity user accounts
5. (Optional.) Managing online identity users
   - Configuring static identity users
   - Specifying the username match mode for user identification
   - Importing online identity users from a remote server
   - Deleting online identity users
6. (Optional.) Deleting identity groups

Enabling the user identification feature

About this task

With the user identification feature, the device learns information about online usersfrom the user access modules. The device uses the obtained information for user identification and works with other security features for identity-based network access control.

Procedure

1. Enter system view.

   system-view

2. Enable the user identification feature.

   user-identity enable

   By default, the user identification feature is disabled.

Configuring remote servers and an identity user import policy

Configuring a RESTful server

About this task

The RESTful server view defines the related parameter settings for the device to communicate with the RESTful server. The parameters include the login account and the URIs of the RESTful server. After establishing a connection with the RESTful server, the device can import identity user accounts, identity groups, and online identity users from the server.

Restrictions and guidelines

The system can have only one RESTful server.

Procedure

1. Enter system view.

   system-view

2. Create a RESTful server and enter its view.

   user-identity restful-server *server-name*

3. Specify the username and password to log in to the RESTful server.

   login-name *user-name* **password** { **cipher** | **simple** } *string*

   By default, no username or password is specified for the device to log in to the RESTful server.

   The specified username and password must exist on the RESTful server. Otherwise, the device cannot establish a connection with the RESTful server.

4. Specify a URI for the RESTful server.

   uri { **get-online-user** | **get-user-database** | **get-user-group-database** | **put-offline-user** | **put-online-user** } *uri-string*

   By default, no URIs are specified for the RESTful server.

   The specified URIs must be the same as those on the RESTful server. Otherwise, user information interaction will fail.

   You can repeat this command to specify multiple URIs of the RESTful server.

5. Specify an MPLS L3VPN instance for the RESTful server.

vpn-instance *vpn-instance-name*

By default, the RESTful server belongs to the public network.

6. (Optional.) Configure RESTful server reachability detection:
   a. Enable RESTful server reachability detection.

      connection-detect enable

      By default, RESTful server reachability detection is disabled.
   b. Configure parameters for RESTful server reachability detection.

      connection-detect { **interval** *interval* | **maximum** *max-times* }

      By default, the reachability detection interval is 5 minutes and the maximum number of probes per detection is 3.

Configuring an LDAP scheme

About this task

An LDAP scheme includes the LDAP server that interacts with the device and the related parameter settings. After establishing a connection with the LDAP server, the device can import identity user accounts and identity groups from the server.

For more information about LDAP attribute maps, see "Configuring AAA."

Restrictions and guidelines

The device cannot import online identity users from the LDAP server.

When importing identity user accounts and identity groups from the LDAP server, the device does not carry parent groups to which the identity users or identity groups belong by default. To configure the device to carry the parent group information, you must configure an LDAP attribute map and specify the LDAP attribute map in the LDAP scheme.

Procedure

1. Enter system view.

   system-view
2. Configure an LDAP server.

   The LDAP server configuration includes the server IP address, the administrator DN and password for binding with the LDAP server, DN search policy, and user group filter. For more information, see "Configuring AAA."
3. Create an LDAP attribute map and enter LDAP attribute map view.

   ldap attribute-map *map-name*
4. Configure an LDAP attribute mapping entry for the user group attribute.

   map ldap-attribute *ldap-attribute-name* [**prefix** *prefix-value* **delimiter** *delimiter-value*] **aaa-attribute user-group**
5. Return to system view.

   quit
6. Create an LDAP scheme and enter LDAP scheme view.

   ldap scheme *ldap-scheme-name*
7. Specify the LDAP authentication server.

   authentication-server *server-name*

   By default, no LDAP authentication server is specified.
8. Specify the LDAP attribute map.

   attribute-map *map-name*

By default, no LDAP attribute map is specified.

After this command is executed, the device imports identity user accounts and identity groups with parent group information. However, the device does not create new identity groups based on the parent group information.

Configuring an identity user import policy

About this task

To import identity user accounts, online identity users, or identity groups from a RESTful server or LDAP servers, you must configure an identity user import policy. In this policy, you can set the parameters for the device to interact with the RESTful server or LDAP servers.

Restrictions and guidelines

The system can have only one identity user import policy. Before you configure a new identity user import policy, you must delete the existing one.

Procedure

1. Enter system view.

   `system-view`

2. Create an identity user import policy and enter its view.

   `user-identity user-import-policy` *`policy-name`*

3. Specify a RESTful server.

   `restful-server` *`server-name`*

   By default, no RESTful server is specified.

   You can specify only one RESTful server. Before you specify a new RESTful server, remove the currently specified one.

4. Specify an LDAP scheme.

   `ldap-scheme` *`ldap-scheme-name`*

   By default, no LDAP schemes are specified.

   You can specify a maximum of 16 LDAP schemes.

5. (Optional.) Set the interval for automatic identity user account import.

   `account-update-interval` *`interval`*

   By default, the interval for automatic identity user account import is 24 hours.

6. Specify the type of user information to be imported from LDAP servers.

   `import-type` `{` **`all`** `|` **`group`** `|` **`user`** `}`

   By default, the device imports both user information and user group information from LDAP servers.

Configuring a security management server set

About this task

The security management server set view defines the related parameters of the device to communicate with security management servers. The parameters include the IP addresses of the servers, the port number for listening to the servers, and the shared key to secure communication between the device and the servers.

When the device establishes connections with security management servers, it can receive user online and offline notifications that the servers push to the device.

- Based on user online notifications, the device obtains online user information and creates online identity user entries for users that match identity user accounts.
- Based on user offline notifications, the device obtains information about users that have gone offline and deletes the corresponding online identity user entries.

Restrictions and guidelines

The system can have only one security management server set.

Procedure

1. Enter system view.
   system-view
2. Create a security management server set and enter its view.
   user-identity security-manage-server *server-set-name*
3. Specify IP addresses of security management servers.
   ip *ip-address*&<1-10>

   By default, no IP addresses of security management servers are specified.
4. Configure the encryption algorithm and shared key for securing communication with security management servers.
   encryption algorithm { **3des** | **aes128** } **key** { **simple** | **cipher** } *string*

   By default, no encryption algorithm or shared key is configured for securing communication with security management servers.
5. Set the port number for listening to security management servers.
   listen-port *port-num*

   By default, the device listens to security management servers on port 8001.

Managing identity user accounts

Enabling automatic identity user account import

About this task

After this feature is enabled, the device first imports all identity user accounts and online identity user information from the servers specified in the identity user import policy. Then, the device periodically imports identity user accounts from the remote servers at the interval set by using the **account-update-interval** command.

Restrictions and guidelines

If automatic identity user account import is enabled but user identity is disabled, the device can import only identity user accounts from the remote servers specified in the policy.

If RESTful server reachability detection is enabled for the RESTful server in the specified policy, the device will automatically import online identity user information from that server when the state of that server changes from unreachable to reachable.

Procedure

1. Enter system view.
   system-view
2. Enable automatic identity user account import.
   user-identity user-account auto-import policy *policy-name*

   By default, automatic identity user account import is disabled.

Manually importing identity user accounts from remote servers

About this task

Perform this task to manually import identity user accounts from remote servers. The device initiates user information requests to the servers, imports network access user account information from the servers, and then creates corresponding identity user accounts. If the device fails to import an account, the device skips the account and continues to import the next account.

Prerequisites

Before you import identity user accounts from remote servers, make sure the following conditions are met:

- An identity user import policy is configured.
- The specified URIs of the RESTful server or the IP addresses of the LDAP servers in the policy must be valid.

Procedure

1. Enter system view.

   system-view

2. Import identity user accounts from remote servers.

   user-identity user-account import policy *policy-name*

Manually importing identity user accounts from a .csv file

About this task

Perform this task to manually import identity user accounts from a .csv file. If the device fails to import an account, it stops importing the remaining identity user accounts in the file.

Prerequisites

Before you import identity user accounts from a .csv file, make sure the .csv file is a standard .csv file. To see the format of a standard .csv file, use the **user-identity user-account export url** command to export a standard template.

Procedure

1. Enter system view.

   system-view

2. Import identity user accounts from a .csv file.

   user-identity user-account import url *url-string* [**vpn-instance** *vpn-instance-name*] [**auto-create-group** | **override** | **start-line** *line-number*] *

Manually exporting identity user accounts

About this task

Perform this task to export identity user accounts on the device to a .csv file. You can directly import identity user accounts in the exported file to other devices. Or, you can edit the exported file and then import identity user accounts in the file back to the device or to other devices.

If you specify the **template** keyword, the device exports a standard .csv file template. You can use this file template as a reference when you edit .csv files.

Procedure

1. Enter system view.

   system-view

2. Export identity user accounts to a .csv file.

   user-identity user-account export url *url-string* [{ **domain** *domain-name* | **null-domain** } [**user** *user-name*] | **template**] [**vpn-instance** *vpn-instance-name*]

Deleting identity user accounts

About this task

Identity user accounts can be deleted by using the following methods:

- **Manual deletion**—The administrator uses a command to delete identity user accounts imported from remote servers or .csv files.
- **Dynamic deletion**—Deletion of network access users in the local user database triggers the deletion of corresponding identity user accounts.

Manually deleting identity user accounts

To manually delete identity user accounts, execute the following command in user view:

reset user-identity user-account { **all** | { **domain** *domain-name* | **null-domain** } [**name** *user-name*] }

Managing online identity users

Configuring static identity users

Restrictions and guidelines

A username can be bound with multiple IP addresses, multiple MAC addresses, or multiple IP-MAC address combinations. However, an IP address, MAC address, or IP-MAC address combination cannot be bound with multiple usernames.

Procedure

1. Enter system view.

   system-view

2. Configure a static identity user.

user-identity static-user *user-name* [**domain** *domain-name*] **bind** { { **ipv4** *ipv4-address* | **ipv6** *ipv6-address* } | **mac** *mac-address* } *

Specifying the username match mode for user identification

About this task

Perform this task to specify the username match mode for user identification. The device creates online identity users only for online users whose usernames can match the usernames in the local identity user accounts.

Restrictions and guidelines

This feature takes effect only on online identity users that access the current device.

Procedure

1. Enter system view.

   `system-view`

2. Specify the username match mode for user identification.

   `user-identity online-user-name-match` { **`keep-original`** | **`with-domain`** | **`without-domain`** }

   By default, the username match mode for user identification is **`keep-original`**.

Importing online identity users from a remote server

About this task

Perform this task to import online identity user information from a remote server. The device initiates a real-time online user information request to the server and then imports all online user information.

Restrictions and guidelines

The device can import online identity users only from an IMC RESTful server.

For the device to successfully import online identity users from a remote server, make sure the user identification feature is enabled.

Prerequisites

Before you import online identity users from a remote server, make sure the following conditions are met:

- An identity user import policy is configured.
- The specified URIs of the RESTful server in the policy must be valid.

Procedure

1. Enter system view.

   `system-view`

2. Import online identity users.

   `user-identity online-user import policy` *`policy-name`*

Deleting online identity users

About this task

Online identity users can be deleted by using the following methods:

- **Manual deletion**—The administrator uses a command to delete dynamic online identity users imported from remote servers and static online identity users.
- **Dynamic deletion**—Multiple events trigger the deletion of identity users.
  - After a network access user goes offline, the access module instructs the user identification module to delete the associated online identity users.
  - After the device reboots, all dynamic online identity users are deleted.
  - After the user identification feature is disabled, all dynamic online identity users are deleted.
  - After a user on a remote server goes offline, the remote server instructs the device to delete the associated online identity user.

Manually deleting dynamic online identity users

To manually delete dynamic online identity users, execute the following command in user view:

reset user-identity dynamic-online-user { **all** | { **domain** *domain-name* | **null-domain** } [**name** *user-name*] | { { **ip** *ipv4-address* | **ipv6** *ipv6-address* } | **mac** *mac-address* } * }

Manually deleting static online identity users

1. Enter system view.

   system-view

2. Delete static online identity users.

   undo user-identity static-user *user-name* [**domain** *domain-name*] [**bind** { { **ipv4** *ipv4-address* | **ipv6** *ipv6-address* } | **mac** *mac-address* } *]

Deleting identity groups

About this task

Identity groups can be deleted by using the following methods:

- **Manual deletion**—The administrator uses a command to delete identity groups imported from remote servers or .csv files.
- **Dynamic deletion**—Deletion of user groups in the local user database triggers the deletion of corresponding identity groups.

Manually deleting identity groups

To manually delete identity groups, execute the following command in user view:

reset user-identity user-group { **all** | { **domain** *domain-name* | **null-domain** } [**name** *group-name*] }

Display and maintenance commands for user identification

Execute **display** commands in any view.

Task	Command
Display information about identity users or identity groups.	**display user-identity** { **domain** *domain-name* | **null-domain** } { **user** [*user-name* [**group**]] | **user-group** [*group-name* [**member** { **group** | **user** }]] }
Display information about active identity groups.	**display user-identity active-user-group** { **all** | **domain** *domain-name* | **null-domain** }
Display information about all identity users or identity groups.	**display user-identity all** { **user** | **user-group** }
Display information about online identity users.	**display user-identity online-user** { **domain** *domain-name* | **null-domain** } **name** *user-name*
Display RESTful server configuration.	**display user-identity restful-server** [*server-name*]
Display configuration information for security	**display user-identity**

Task	Command
management server sets.	**security-manage-server** [*server-set-name*]
Display identity user import policy information.	**display user-identity user-import-policy** [*policy-name*]

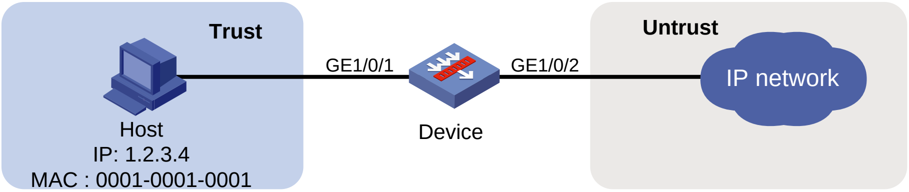
User identification configuration examples

Example: Configuring static user identification

Network configuration

As shown in Figure 2, the administrator permits user **usera** with IP address 1.2.3.4 and MAC address 0001-0001-0001 to access the network without identity authentication. Configure a security policy for access control, which permits the user to access the network only from 8:00 to 18:00 in working days.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.2.3.5 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 3.1.1.2 24 2.2.2.2
   ```

4. Configure user identification:

   # Create a network access user named **usera**.

   ```
   [Device] local-user usera class network
   [Device-luser-network-usera] quit
   ```

# Configure a static identity user named **usera** with IP address 1.2.3.4 and MAC address 0001-0001-0001.

```
[Device] user-identity static-user usera bind ipv4 1.2.3.4 mac 0001-0001-0001
```

# Enable the user identification feature.

```
[Device] user-identity enable
```

5. Configure time range and security policy settings:

   # Create a time range named **work** and specify the time range in working days.

   ```
   [Device] time-range work 08:00 to 18:00 working-day
   ```

   # Configure a rule named **ippolicy1** in the IPv4 security policy to permit user **usera** to access the IP network only from 8:00 to 18:00 in working days.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name ippolicy1
   [Device-security-policy-ip-1-ippolicy1] source-zone trust
   [Device-security-policy-ip-1-ippolicy1] destination-zone untrust
   [Device-security-policy-ip-1-ippolicy1] action pass
   [Device-security-policy-ip-1-ippolicy1] user usera
   [Device-security-policy-ip-1-ippolicy1] time-range work
   [Device-security-policy-ip-1-ippolicy1] quit
   [Device-security-policy-ip] quit
   ```

Verifying the configuration

# Display online identity user information for user **usera**.

```
[Device] display user-identity online-user null-domain name usera
User name: usera
  IP  : 1.2.3.4
  MAC : 0001-0001-0001
  Type: Static

Total 1 records matched
```

# Verify that the user can access the network only from 8:00 to 18:00 in working days. (Details not shown.)

Contents

Configuring password control

About password control

Password control allows you to implement the following features:

- Manage login and super password setup, expirations, and updates for device management users.
- Control user login status based on predefined policies.

For more information about local users, see "Configuring AAA." For information about super passwords, see RBAC in *Fundamentals Configuration Guide*.

Password setting

Minimum password length

You can define the minimum length of user passwords. The system rejects the setting of a password that is shorter than the configured minimum length.

Password composition policy

A password can be a combination of characters from the following types:

- Uppercase letters A to Z.
- Lowercase letters a to z.
- Digits 0 to 9.
- Special characters in Table 1.

Table 1 Special Characters

Character name	Symbol	Character name	Symbol
Ampersand sign	&	Apostrophe	'
Asterisk	*	At sign	@
Back quote	`	Back slash	\
Blank space	N/A	Caret	^
Colon	:	Comma	,
Dollar sign	$	Dot	.
Equal sign	=	Exclamation point	!
Left angle bracket	<	Left brace	{
Left bracket	[	Left parenthesis	(
Minus sign	-	Percent sign	%
Plus sign	+	Pound sign	#
Quotation marks	"	Right angle bracket	>
Right brace	}	Right bracket	]
Right parenthesis	)	Semi-colon	;

Character name	Symbol	Character name	Symbol
Slash	/	Tilde	~
Underscore	_	Vertical bar	|

Depending on the system's security requirements, you can set the minimum number of character types a password must contain and the minimum number of characters for each type, as shown in Table 2.

Table 2 Password composition policy

Password combination level	Minimum number of character types	Minimum number of characters for each type
Level 1	One	One
Level 2	Two	One
Level 3	Three	One
Level 4	Four	One

When a user sets or changes a password, the system checks if the password meets the combination requirement. If it does not, the operation fails.

Password complexity checking policy

A less complicated password is more likely to be cracked, such as a password containing the username or repeated characters. For higher security, you can configure a password complexity checking policy to ensure that all user passwords are relatively complicated. When a user configures a password, the system checks the complexity of the password. If the password is complexity-incompliant, the configuration will fail.

You can apply the following password complexity requirements:

- A password cannot contain the username or the reverse of the username. For example, if the username is **abc**, a password such as **abc982** or **2cba** is not complex enough.
- A minimum of three identical consecutive characters is not allowed. For example, password **a111** is not complex enough.

Password updating and expiration

Password updating

This feature allows you to set the minimum interval at which users can change their passwords. A user can only change the password once within the specified interval.

The minimum interval does not apply to the following situations:

- A user is prompted to change the password at the first login.
- The password aging time expires.

Password expiration

Password expiration imposes a lifecycle on a user password. After the password expires, the user needs to change the password.

The system displays an error message for a login attempt with an expired password. The user is asked to provide a new password. The new password must be valid, and the user must enter exactly the same password when confirming it.

Web users, Telnet users, SSH users, and console users can change their own passwords. FTP users must have their passwords changed by the administrator.

Early notice on pending password expiration

When a user logs in, the system checks whether the password will expire in a time equal to or less than the specified notification period. If so, the system notifies the user when the password will expire and provides a choice for the user to change the password.

- If the user sets a new valid password, the system records the new password and the setup time.
- If the user does not or fails to change the password, the system allows the user to log in by using the current password until the password expires.

Web users, Telnet users, SSH users, and console users can change their own passwords. FTP users must have their passwords changed by the administrator.

Login with an expired password

You can allow a user to log in a certain number of times within a period of time after the password expires. For example, if you set the maximum number of logins with an expired password to 3 and the time period to 15 days, a user can log in three times within 15 days after the password expires.

Password history

This feature allows the system to store passwords that a user has used.

When a network access user changes the password, the system compares the new password with the current password and those stored in the password history records. The new password must be different from the current one and those stored in the history records by a minimum of four different characters. Otherwise, the system will display an error message, and the new password is not successfully set.

The local passwords and super passwords for device management users are stored in hash form and cannot be converted to plain texts. When a device management user changes a local password or super password, follow these rules:

- If the new password is set by using the hash method, the system will not compare the new password with the current one and those stored in the history password records.
- If the new password is set in plain text, the system compares the new password with the current password and those stored in the password history records. A new password must be different from those stored in the history password records. If the current password is required, the new password must also be different from the current one by a minimum of four different characters. Otherwise, the system will display an error message, and the new password is not successfully set.

You can set the maximum number of history password records for the system to maintain for each user. When the number of history password records exceeds the setting, the most recent record overwrites the earliest one.

Current login passwords are not stored in the password history for device management users. Device management users have their passwords saved in cipher text, which cannot be recovered to plaintext passwords.

User login control

First login

By default, if the global password control feature is enabled, users must change the password at first login before they can access the system. In this situation, password changes are not subject to the minimum password update interval. If it is not necessary for users to change the password at first login, disable the password change at first login feature.

Password control blacklist

The password control blacklist prevents abnormal users from logging in by recording the login failures and maintaining the status of blacklisted user accounts.

The system adds the information of the following users that fail to log in to the password control blacklist:

- FTP, Web, or VTY users.
- Users logging in to the device through console ports.

The system does not add the user accounts of nonexistent users (users not configured on the device) to the password control blacklist if they fail to log in.

The device supports the following recording modes for adding the user information to the blacklist for users failing authentication:

- **Username only**—Adds only usernames to the blacklist. In this mode, a user account matches a blacklist entry as long as the username matches the entry.
- **Username and IP address**—Adds both usernames and IP addresses to the blacklist. In this mode, a user account matches a blacklist entry only when both the username and the login IP address match the entry.

The device will create a blacklist entry for each IP address for a user account when the following conditions are both met:

- Both usernames and IP addresses are added to the password control blacklist.
- A user uses the same user account to log in to the device from different IP addresses and fails the logins.

When the maximum number of blacklist entries for the user account is reached, a blacklist entry for a new IP address of the user account will overwrite the earliest blacklist entry for the user account.

Login attempt limit

Limiting the number of consecutive login failures can effectively prevent password guessing.

When the user fails the maximum number of consecutive attempts, login attempt limit limits the user and user account in any of the following ways:

- Locks the user account and the user's IP address permanently. No users can use this account to log in to the device from this IP address unless the account is manually removed from the password control blacklist.
- Allows the user to continue using the user account. The user's IP address and user account are removed from the password control blacklist when the user uses this account to successfully log in to the device.
- Locks the user account and the user's IP address for a period of time.

  The user can use the account to log in from the IP address when either of the following conditions exists:

  - The locking timer expires.
  - The account is manually removed from the password control blacklist before the locking timer expires.

NOTE:

This account is locked only for the user at the locked IP address. A user from an unlocked IP address can still use this account, and the user at the locked IP address can use other unlocked user accounts.

Maximum account idle time

You can set the maximum account idle time for user accounts. When an account is idle for this period of time since the last successful login, the account becomes invalid.

Login control with a weak password

The system checks for weak passwords for Telnet, SSH, HTTP, or HTTPS device management users. A password is weak if it does not meet the following requirements:

- Password composition restriction.
- Minimum password length restriction.
- Password complexity checking policy.

By default, the system displays a message about a weak password but does not force the user to change it. To improve the device security, you can enable the mandatory weak password change feature, which forces the users to change the identified weak passwords. The users can log in to the device only after their passwords meet the password requirements.

First login with a default username and password

The factory defaults contain a default username and password. If the device starts up with factory defaults, whether the system checks the default password depends on the device model.

Password not displayed in any form

For security purposes, nothing is displayed when a user enters a password.

Logging

The system generates a log each time a user changes its password successfully or is added to the password control blacklist because of login failures.

Restrictions and guidelines: Password control configuration

IMPORTANT:

To successfully enable the global password control feature and allow device management users to log in to the device, make sure the device have sufficient storage space.

The password control features can be configured in several different views, and different views support different features. The settings configured in different views or for different objects have the following application ranges:

- Settings for super passwords apply only to super passwords.
- Settings in local user view apply only to the password of the local user.
- Settings in user group view apply to the passwords of the local users in the user group if you do not configure password policies for these users in local user view.
- Global settings in system view apply to the passwords of the local users in all user groups if you do not configure password policies for these users in both local user view and user group view.

For local user passwords, the settings with a smaller application scope have higher priority.

Password control tasks at a glance

To configure password control, perform the following tasks:

1. Enabling password control
2. (Optional.) Setting global password control parameters
3. (Optional.) Setting user group password control parameters
4. (Optional.) Setting local user password control parameters
5. (Optional.) Setting super password control parameters

Enabling password control

About this task

The password composition policy, minimum password length, and username checking features are independent of the global password control feature. Other password control features (such as password expiration or password history management) can take effect only after the global password control feature is also enabled.

Restrictions and guidelines

After global password control is enabled, follow these restrictions and guidelines:

- You cannot display the password and super password configurations for device management users by using the corresponding **display** commands.
- You cannot display the password configuration for network access users by using the corresponding **display** command.
- The passwords configured for local users must contain a minimum of four different characters.
- To ensure correct function of password control, configure the device to use NTP to obtain the UTC time. After global password control is enabled, password control will record the UTC time when the password is set. The recorded UTC time might not be consistent with the actual UTC time due to power failure or device reboot. The inconsistency will cause the password expiration feature to malfunction. For information about NTP, see *Network Management and Monitoring Configuration Guide*.
- The device automatically generates a .dat file and saves the file to the storage media. The file is used to record authentication and login information of the local users. Do not manually delete or modify the file.

Procedure

1. Enter system view.

   system-view

2. Enable the global password control feature.

   password-control enable [**network-class**]

   By default, the global password control feature is disabled for device management and network access users.

3. (Optional.) Enable a specific password control feature.

   password-control { **aging** | **composition** | **history** | **length** } **enable**

   By default, all four password restriction features are enabled.

Setting global password control parameters

Restrictions and guidelines

The global password control parameters in system view apply to all device management and network access local users.

You can configure all password control features for device management users. The password aging time, minimum password length, password complexity policy, password composition policy, and user login attempt limit can be configured in system view, user group view, and local user view.

You can configure only the following password control features for network access users:

- Minimum password length.
- Password complexity policy.
- Password composition policy.

- Minimum password update interval.
- Maximum number of history password records for each user.

Where, the minimum password length, password complexity policy, and password composition policy can be configured in system view, user group view, and local user view.

The password settings with a smaller application scope have higher priority. For local users, password settings configured in local user view have the highest priority, and global settings in system view have the lowest priority.

The global password control feature enables the system to record history passwords. When the number of history password records of a user reaches the maximum number, the newest history record overwrites the earliest one. To delete the existing history password records, use one of the following methods:

- Use the **undo password-control enable** command to disable the password control feature globally.
- Use the **reset password-control history-record** command to clear the passwords manually.

The **password-control login-attempt** command takes effect immediately and can affect the users already in the password control blacklist. If the user information items to add to the password control blacklist change, the system will clear the password control blacklist and restart the recording. Other password control configurations do not take effect on users that have been logged in or passwords that have been configured.

Procedure

1. Enter system view.

   system-view

2. Configure password settings.
   - Set the minimum password length.

     password-control length *length*

     The default length is 10 characters.
   - Configure the password composition policy.

     password-control composition type-number *type-number* [**type-length** *type-length*]

     By default, a password must contain a minimum of two character types and a minimum of one character for each type.
   - Configure the password complexity checking policy.

     password-control complexity { **same-character** | **user-name** } **check**

     By default, the username checking is enabled but repeated character checking is disabled.
   - Set the maximum number of history password records for each user.

     password-control history *max-record-number*

     The default setting is 4.

3. Configure password updating and expiration.
   - Set the minimum password update interval.

     password-control update interval *interval*

     The default setting is 24 hours.
   - Set the password aging time.

     password-control aging *aging-time*

     The default setting is 90 days.
   - Set the number of days during which a user is notified of the pending password expiration.

password-control alert-before-expire *alert-time*

The default setting is 7 days.

- Set the maximum number of days and maximum number of times that a user can log in after the password expires.

  password-control expired-user-login delay *delay* **times** *times*

  By default, a user can log in three times within 30 days after the password expires.

4. Configure user login control.
   - Set the maximum number of blacklist entries for a user account.

     password-control per-user blacklist-limit *max-number*

     By default, the maximum number of blacklist entries for a user account is 32.
   - Configure the login attempt limit.

     password-control login-attempt *login-times* [**exceed** { **lock** | **lock-time** *time* | **unlock** }]

     By default, the maximum number of login attempts is 3 and a user failing to log in after the specified number of attempts must wait for 1 minute before trying again.
   - Add only usernames to the password control blacklist for users failing authentication.

     password-control blacklist user-info username-only

     By default, both usernames and IP addresses are added to the password control blacklist when the users fail authentication.
   - Set the maximum account idle time.

     password-control login idle-time *idle-time*

     The default setting is 90 days.

     If a user account is idle for this period of time, the account becomes invalid and can no longer be used to log in to the device. To disable the account idle time restriction, set the idle time value to 0.
   - Disable password change at first login.

     undo password-control change-password first-login enable

     By default, the password change at first login is enabled.
   - Enable mandatory weak password change.

     password-control change-password weak-password enable

     By default, the mandatory weak password change feature is disabled.

Setting user group password control parameters

1. Enter system view.

   system-view
2. Create a user group and enter its view.

   user-group *group-name*

   For information about how to configure a user group, see "Configuring AAA."
3. Configure the password aging time for the user group.

   password-control aging *aging-time*

   By default, the password aging time of the user group equals the global password aging time.
4. Configure the minimum password length for the user group.

   password-control length *length*

By default, the minimum password length of the user group equals the global minimum password length.

5. Configure the password composition policy for the user group.

   password-control composition type-number *type-number* [**type-length** *type-length*]

   By default, the password composition policy of the user group equals the global password composition policy.

6. Configure the password complexity checking policy for the user group.

   password-control complexity { **same-character** | **user-name** } **check**

   By default, the password complexity checking policy of the user group equals the global password complexity checking policy.

7. Configure the login attempt limit.

   password-control login-attempt *login-times* [**exceed** { **lock** | **lock-time** *time* | **unlock** }]

   By default, the login-attempt policy of the user group equals the global login-attempt policy.

Setting local user password control parameters

1. Enter system view.

   system-view

2. Create a device management or network access user and enter its view.
   - Create a device management user and enter its view.

     local-user *user-name* **class manage**
   - Create a network access user and enter its view.

     local-user *user-name* **class network**

   For information about how to configure a local user, see "Configuring AAA."

3. Configure the password aging time for the local user.

   password-control aging *aging-time*

   By default, the setting equals that for the user group to which the local user belongs. If no aging time is configured for the user group, the global setting applies to the local user.

   This command is available only for device management users.

4. Configure the minimum password length for the local user.

   password-control length *length*

   By default, the setting equals that for the user group to which the local user belongs. If no minimum password length is configured for the user group, the global setting applies to the local user.

5. Configure the password composition policy for the local user.

   password-control composition type-number *type-number* [**type-length** *type-length*]

   By default, the settings equal those for the user group to which the local user belongs. If no password composition policy is configured for the user group, the global settings apply to the local user.

6. Configure the password complexity checking policy for the local user.

   password-control complexity { **same-character** | **user-name** } **check**

   By default, the settings equal those for the user group to which the local user belongs. If no password complexity checking policy is configured for the user group, the global settings apply to the local user.

7. Configure the login attempt limit.

   password-control login-attempt *login-times* [**exceed** { **lock** | **lock-time** *time* | **unlock** }]

   By default, the settings equal those for the user group to which the local user belongs. If no login-attempt policy is configured for the user group, the global settings apply to the local user.

   This command is available only for device management users.

Setting super password control parameters

1. Enter system view.

   system-view

2. Set the password aging time for super passwords.

   password-control super aging *aging-time*

   The default setting is 90 days.

3. Configure the minimum length for super passwords.

   password-control super length *length*

   The default setting is 10 characters.

4. Configure the password composition policy for super passwords.

   password-control super composition type-number *type-number* [**type-length** *type-length*]

   By default, a super password must contain a minimum of two character types and a minimum of one character for each type.

Display and maintenance commands for password control

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display password control configuration.	**display password-control** [**super**]
Display information about users in the password control blacklist.	**display password-control blacklist** [**user-name** *user-name* | **ip** *ipv4-address* | **ipv6** *ipv6-address*]
Delete users from the password control blacklist.	**reset password-control blacklist** [**user-name** *user-name*]
Clear history password records.	**reset password-control history-record** [**user-name** *user-name* | **super** [**role** *role name*] | **network-class** [**user-name** *user-name*]]

Contents

Configuring portal authentication

About portal authentication

Portal authentication controls user access to networks. Portal authenticates a user by the username and password the user enters on a portal authentication page. Typically, portal authentication is deployed on the access layer and vital data entries.

In a portal-enabled network, users can actively initiate portal authentication by visiting the authentication website provided by the portal Web server. Or, they are redirected to the portal authentication page for authentication when they visit other websites.

The device supports Portal 1.0, Portal 2.0, and Portal 3.0.

Advantages of portal authentication

Portal authentication has the following advantages:

- Allows users to perform authentication through a Web browser without installing client software.
- Provides ISPs with diversified management choices and extended functions. For example, the ISPs can place advertisements, provide community services, and publish information on the authentication page.
- Supports multiple authentication modes. For example, re-DHCP authentication implements a flexible address assignment scheme and saves public IP addresses. Cross-subnet authentication can authenticate users who reside in a different subnet than the access device.

Extended portal functions

By forcing patching and anti-virus policies, extended portal functions help hosts to defend against viruses. Portal supports the following extended functions:

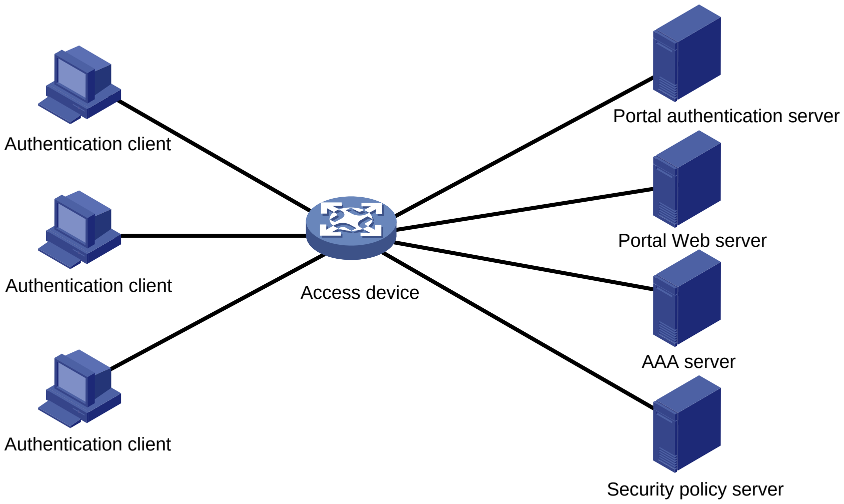
- **Security check**—Detects after authentication whether or not a user host installs anti-virus software, virus definition file, unauthorized software, and operating system patches.
- **Resource access restriction**—Allows an authenticated user to access certain network resources such as the virus server and the patch server. Users can access more network resources after passing security check.

Security check must cooperate with the IMC security policy server and the iNode client.

Portal system

A typical portal system consists of these basic components: authentication client, access device, portal authentication server, portal Web server, AAA server, and security policy server.

Figure 1 Portal system



Authentication client

An authentication client is a Web browser that runs HTTP/HTTPS or a user host that runs a portal client. Security check for the user host is implemented through the interaction between the portal client and the security policy server. Only the iNode client is supported.

Access device

An access device provides access services. It has the following functions:

- Redirects all HTTP or HTTPS requests of unauthenticated users to the portal Web server.
- Interacts with the portal authentication server and the AAA server to complete authentication, authorization, and accounting.
- Allows users that pass portal authentication to access authorized network resources.

Portal server

A portal server collectively refers to a portal authentication server and portal Web server.

The portal Web server pushes the Web authentication page to authentication clients and forwards user authentication information (username and password) to the portal authentication server. The portal authentication server receives authentication requests from authentication clients and interacts with the access device to authenticate users. The portal Web server is typically integrated with the portal authentication server and it can also be an independent server.

AAA server

The AAA server interacts with the access device to implement authentication, authorization, accounting for portal users. In a portal system, a RADIUS server can perform authentication, authorization, accounting for portal users, and an LDAP server can perform authentication for portal users.

Security policy server

The security policy server interacts with the portal client and the access device for security check and authorization for users. Only hosts that run portal clients can interact with the security policy server.

Portal authentication using a remote portal server

The components of a portal system interact as follows:

1. An unauthenticated user initiates authentication by accessing an Internet website through a Web browser. When receiving the HTTP or HTTPS request, the access device redirects it to the

Web authentication page provided by the portal Web server. The user can also visit the authentication website to log in. The user must log in through the iNode client for extended portal functions.

2. The user enters the authentication information on the authentication page/dialog box and submits the information. The portal Web server forwards the information to the portal authentication server. The portal authentication server processes the information and forwards it to the access device.
3. The access device interacts with the AAA server to implement authentication, authorization, accounting for the user.
4. If security policies are not imposed on the user, the access device allows the authenticated user to access networks.

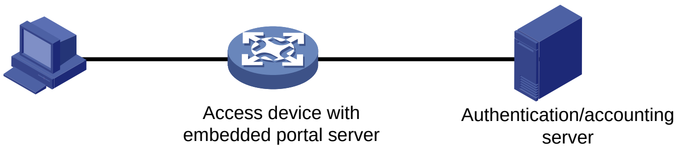
   If security policies are imposed on the user, the portal client, the access device, and the security policy server interact to check the user host. If the user passes the security check, the security policy server authorizes the user to access resources based on the check result.

Local portal service

System components

As shown in Figure 2, a local portal system consists of an authentication client, access device, and AAA server. The access device acts as both the portal Web server and the portal authentication server to provide the local portal Web service for the authentication client. The authentication client can only be a Web browser, and it cannot be a user host that runs a portal client. Therefore, extended portal functions are not supported and no security policy server is required.

Figure 2 System components



Portal page customization

To provide the local portal web service, you must customize a set of authentication pages that the device will push to users. You can customize multiple sets of authentication pages, compress each set of the pages to a .zip file, and upload the compressed files to the storage medium of the device. On the device, you must specify one of the files as the default authentication page file by using the **`default-logon-page`** command.

For more information about authentication page customization, see "Customizing authentication pages."

Portal authentication modes

Portal authentication has three modes: direct authentication, re-DHCP authentication, and cross-subnet authentication. In direct authentication and re-DHCP authentication, no Layer 3 forwarding devices exist between the authentication client and the access device. In cross-subnet authentication, Layer 3 forwarding devices can exist between the authentication client and the access device.

Direct authentication

A user manually configures a public IP address or obtains a public IP address through DHCP. Before authentication, the user can access only the portal Web server and predefined authentication-free websites. After passing authentication, the user can access other network resources. The process of direct authentication is simpler than that of re-DHCP authentication.

Re-DHCP authentication

Before a user passes authentication, DHCP allocates an IP address (a private IP address) to the user. The user can access only the portal Web server and predefined authentication-free websites. After the user passes authentication, DHCP reallocates an IP address (a public IP address) to the user. The user then can access other network resources. No public IP address is allocated to users who fail authentication. Re-DHCP authentication saves public IP addresses. For example, an ISP can allocate public IP addresses to broadband users only when they access networks beyond the residential community network.

Only the iNode client supports re-DHCP authentication. IPv6 portal authentication does not support the re-DHCP authentication mode.

Cross-subnet authentication

Cross-subnet authentication is similar to direct authentication, except it allows Layer 3 forwarding devices to exist between the authentication client and the access device.

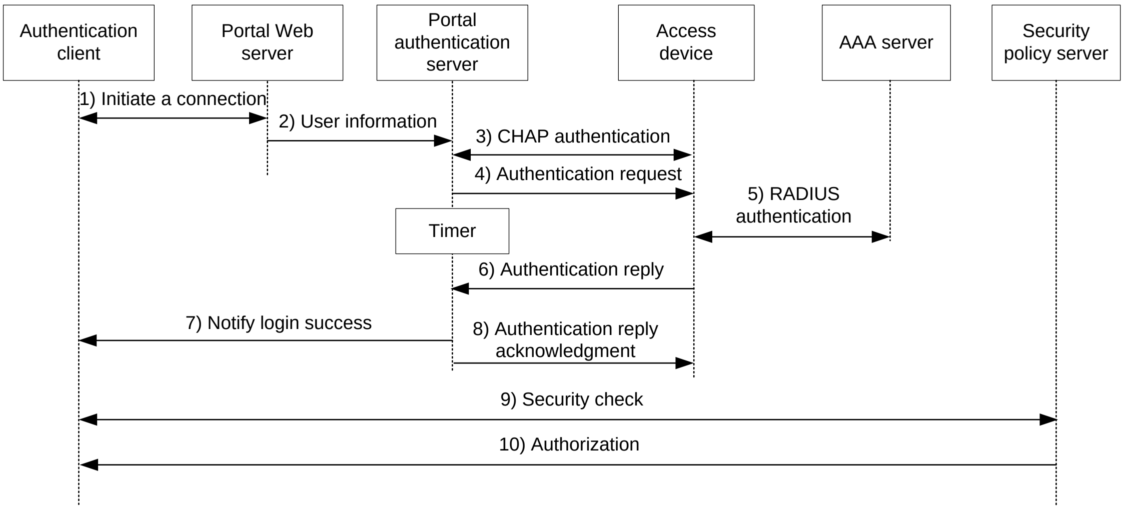
In direct authentication, re-DHCP authentication, and cross-subnet authentication, a user's IP address uniquely identifies the user. After a user passes authentication, the access device generates an ACL for the user based on the user's IP address to control forwarding of the packets from the user. Because no Layer 3 forwarding device exists between authentication clients and the access device in direct authentication and re-DHCP authentication, the access device can learn the user MAC addresses. The access device can enhance its capability of controlling packet forwarding by using the learned MAC addresses.

Portal authentication process

Direct authentication and cross-subnet authentication share the same authentication process. Re-DHCP authentication has a different process as it has two address allocation procedures.

Direct authentication/cross-subnet authentication process (with CHAP/PAP authentication)

Figure 3 Direct authentication/cross-subnet authentication process



The direct/cross-subnet authentication process is as follows:

1. A portal user access the Internet through HTTP or HTTPS, and the HTTP or HTTPS packet arrives at the access device.
   - If the packet matches a portal free rule, the access device allows the packet to pass.
   - If the packet does not match any portal-free rule, the access device redirects the packet to the portal Web server. The portal Web server pushes the Web authentication page to the user for him to enter his username and password.

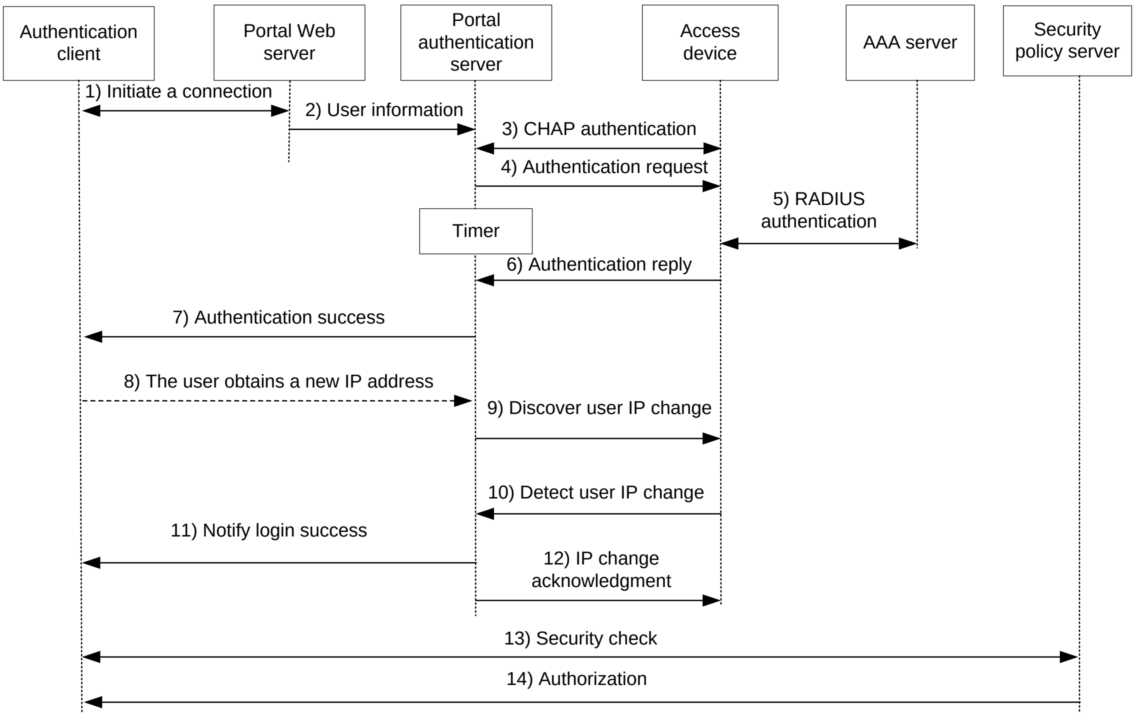
2. The portal Web server submits the user authentication information to the portal authentication server.
3. The portal authentication server and the access device exchange CHAP messages. This step is skipped for PAP authentication. The portal authentication server decides the method (CHAP or PAP) to use.
4. The portal authentication server adds the username and password into an authentication request packet and sends it to the access device. Meanwhile, the portal authentication server starts a timer to wait for an authentication reply packet.
5. The access device and the RADIUS server exchange RADIUS packets.
6. The access device sends an authentication reply packet to the portal authentication server to notify authentication success or failure.
7. The portal authentication server sends an authentication success or failure packet to the client.
8. If the authentication is successful, the portal authentication server sends an authentication reply acknowledgment packet to the access device.

If the client is an iNode client, the authentication process includes step 9 and step 10 for extended portal functions. Otherwise the authentication process is complete.

9. The client and the security policy server exchange security check information. The security policy server detects whether or not the user host installs anti-virus software, virus definition files, unauthorized software, and operating system patches.
10. The security policy server authorizes the user to access certain network resources based on the check result. The access device saves the authorization information and uses it to control access of the user.

Re-DHCP authentication process (with CHAP/PAP authentication)

Figure 4 Re-DHCP authentication process



The re-DHCP authentication process is as follows:

Step 1 through step 7 are the same as those in the direct authentication/cross-subnet authentication process.

8. After receiving the authentication success packet, the client obtains a public IP address through DHCP. The client then notifies the portal authentication server that it has a public IP address.
9. The portal authentication server notifies the access device that the client has obtained a public IP address.
10. The access device detects the IP change of the client through DHCP and then notifies the portal authentication server that it has detected an IP change of the client IP.
11. After receiving the IP change notification packets sent by the client and the access device, the portal authentication server notifies the client of login success.
12. The portal authentication server sends an IP change acknowledgment packet to the access device.

Step 13 and step 14 are for extended portal functions.

13. The client and the security policy server exchanges security check information. The security policy server detects whether or not the user host installs anti-virus software, virus definition files, unauthorized software, and operating system patches.
14. The security policy server authorizes the user to access certain network resources based on the check result. The access device saves the authorization information and uses it to control access of the user.

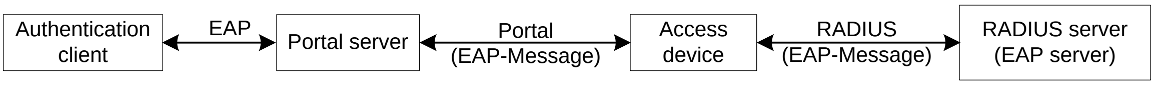
Portal support for EAP

To use portal authentication that supports EAP, the portal authentication server and client must be the IMC portal server and the iNode portal client. Local portal authentication does not support EAP authentication.

Compared with username and password based authentication, digital certificate-based authentication ensures higher security.

The Extensible Authentication Protocol (EAP) supports several digital certificate-based authentication methods, for example, EAP-TLS. Working together with EAP, portal authentication can implement digital certificate-based user authentication.

Figure 5 Portal support for EAP working flow diagram



As shown in Figure 5, the authentication client and the portal authentication server exchange EAP authentication packets. The portal authentication server and the access device exchange portal authentication packets that carry the EAP-Message attributes. The access device and the RADIUS server exchange RADIUS packets that carry the EAP-Message attributes. The RADIUS server that supports the EAP server function processes the EAP packets encapsulated in the EAP-Message attributes, and provides the EAP authentication result.

The access device does not process but only transports EAP-Message attributes between the portal authentication server and the RADIUS server. Therefore, the access device requires no additional configuration to support EAP authentication.

Portal filtering rules

The access device uses portal filtering rules to control user traffic forwarding.

Based on the configuration and authentication status of portal users, the device generates the following categories of portal filtering rules:

- **Category-1**—The rule permits user packets that are destined for the portal Web server and packets that match the portal-free rules to pass through.

- **Category-2**—For an authenticated user with no ACL authorized, the rule allows the user to access any destination network resources. For an authenticated user with an ACL authorized, the rule allows users to access resources permitted by the ACL. The device adds the rule when a user comes online and deletes the rule when the user goes offline.

  The device supports the following types of authorization ACLs:

  - Basic ACLs (ACL 2000 to ACL 2999).
  - Advanced ACLs (ACL 3000 to ACL 3999).

  For an authorization ACL to take effect, make sure the ACL exists and has ACL rules excluding rules configured with the **`counting`**, **`established`**, **`fragment`**, **`source-mac`**, or **`logging`** keyword. For more information about ACL rules, see ACL commands in *ACL and QoS Command Reference*.
- **Category-3**—The rule redirects all HTTP or HTTPS requests from unauthenticated users to the portal Web server.
- **Category-4**—For direct authentication and cross-subnet authentication, the rule forbids any user packets to pass through. For re-DHCP authentication, the device forbids user packets with private source addresses to pass.

After receiving a user packet, the device compares the packet against the filtering rules from category-1 to category-4. Once the packet matches a rule, the matching process completes.

MAC-based quick portal authentication

MAC-based quick portal authentication is applicable to scenarios where users access the network frequently. It allows users to pass authentication without entering a username and password. MAC-based quick portal authentication is also called MAC-trigger authentication or transparent portal authentication.

A MAC binding server is required for MAC-trigger authentication. The MAC binding server records the MAC-to-account bindings of portal users for authentication. The account contains the portal authentication information of the user, including username and password.

Only IPv4 direct authentication supports MAC-based quick portal authentication.

The authentication is implemented as follows:

1. When a user accesses the network for the first time, the access device generates a MAC-trigger entry that records the user's MAC address and access interface. The user can access the network without performing portal authentication if the user's network traffic is below the free-traffic threshold.
2. When the user's network traffic reaches the threshold, the access device sends a MAC binding query to the MAC binding server.
3. The MAC binding server checks whether the MAC address of the user is bound with a portal user account.
   - If a matching MAC-account binding exists, the MAC binding server sends the user authentication information to the access device to initiate portal authentication. The user is authenticated without entering the username and password.
     - If the user fails portal authentication, an authentication failure message is returned to the user. The MAC-trigger entry of the user on the access device is deleted when the entry ages out.
     - If the user passes portal authentication, the access device deletes the MAC-trigger entry of the user.
   - If no matching MAC-account binding exists, the MAC binding server notifies the access device to perform normal portal authentication for the user.
     - If the user fails portal authentication, an authentication failure message is returned to the user. The whole process is finished.

– If the user passes portal authentication, the access device sends the user's MAC address and authentication information to the MAC binding server for MAC-account binding. Additionally, the access device deletes the MAC-trigger entry of the user.

NOTE:

- For information about MAC binding server configuration, see the user manual of the server.

Portal support for NAT444

On a network where portal is used in conjunction with NAT444, portal and NAT444 cooperate as follows:

1. After a portal user passes AAA authentication and is assigned a private IP address, portal notifies NAT444 of the user login.
2. The NAT444 gateway assigns a public IP address and a port block to the portal user. Then, it sends the user private address, public address, and port block mapping to portal.

   On the NAT444 gateway, if no public IP address is available to allocate to the user, portal logs out the user.
3. Portal records the mapping, and reports the mapping to the AAA server. Then, the portal user can use the public IP address and port block to access the external network.

Through the portal and NAT444 cooperation, the AAA server can obtain and maintain NAT mapping information for all portal users, which facilitates user tracing.

To use portal in conjunction with NAT444, specify the user address type as private IPv4 address in the authentication ISP domain for portal users by using the **`user-address-type private-ipv4`** command. For more information about specifying the user address type in an ISP domain, see "Configuring AAA." For more information about NAT444, see NAT in *Layer 3—IP Services Configuration Guide*.

Restrictions and guidelines: Portal configuration

In portal authentication, the device can redirect users' HTTPS requests to the portal Web server. You can use one of the following local certificates for the HTTPS redirect service according to the security requirements and the configuration complexity:

- **Self-signed certificate**—Using this type of certificate is simple in configuration but has low security. You do not need to associate an SSL server policy with the HTTPS redirect service and the default SSL parameters are used. However, a self-signed certificate is not trusted by the browser. When the device redirects HTTPS requests to the portal Web server, a certificate security warning prompt might appear on the browser. If you accept the security risks stated in the prompt, you can ignore the prompt to browse the page.
- **CA-signed certificate**—Using this type of certificate is complex in configuration but has high security. You must obtain a CA certificate, request a local certificate from the CA, create an SSL server policy (with policy name of **https_redirect**), and associate the SSL server policy with the HTTPS redirect service.

For more information about digital certificates, see "Configuring PKI." For more information about the SSL server policy configuration, see "Configuring SSL."

Portal authentication through Web does not support security check for users. To implement security check, the client must be the iNode client.

Portal authentication supports NAT traversal whether it is initiated by a Web client or an iNode client. NAT traversal must be configured when the portal client is on a private network and the portal server is on a public network.

Portal authentication tasks at a glance

To configure portal authentication, perform the following tasks:

1. Configuring a remote portal service

   Perform this task if a remote portal server is used.

   - Configuring a remote portal authentication server
   - Configuring a portal Web server
2. Configuring a local portal service

   Perform this task if the access device acts as a portal authentication server and portal Web server.

   - Configuring the local portal service features
   - Configuring a portal Web server
3. Enabling portal authentication and specifying a portal Web server

   Choose the options to configure on an interface.

   - Enabling portal authentication on an interface
   - Specifying a portal Web server on an interface
4. (Optional.) Configure parameters for preauthentication portal users
   - Configuring a portal preauthentication domain
   - Specifying a preauthentication IP address pool
5. (Optional.) Specifying a portal authentication domain
6. (Optional.) Controlling portal user access
   - Configuring a portal-free rule
   - Configuring an authentication source subnet
   - Configuring an authentication destination subnet
   - Configuring a portal-forbidden rule
   - Specifying port numbers of Web proxy servers
   - Setting the maximum number of portal users
   - Enabling strict-checking on portal authorization information
   - Allowing only users with DHCP-assigned IP addresses to pass portal authentication
   - Blocking portal users that fail portal authentication
   - Configuring support of portal authentication for dual stack
   - Enabling intra-VLAN roaming for portal users
   - Enabling outgoing packets filtering
   - Configuring the portal fail-permit feature
7. (Optional.) Configuring portal detection features
   - Configuring online detection of portal users
   - Configuring portal authentication server detection
   - Configuring portal Web server detection
   - Enabling DHCP packet capture
   - Configuring portal user synchronization
8. (Optional.) Configuring attributes for portal packets and RADIUS packets
   - Configuring portal packet attributes

     You can configure the BAS-IP or BAS-IPv6 attribute for portal packets and specify the device ID.

   - Configuring attributes for RADIUS packets

     You can configure the NAS-Port-Id attribute and apply a NAS-ID profile to an interface.

9. (Optional.) Configuring MAC-based quick portal authentication
   a. Configuring a remote MAC binding server
   b. Configuring a local MAC binding server
   c. Specifying a MAC binding server on an interface
   d. Configure cloud MAC-trigger authentication
10. (Optional.) Configuring online and offline related features for portal users
    - Logging out online portal users
11. (Optional.) Configuring extended portal authentication features
    - Setting the user traffic backup threshold
    - Disabling the Rule ARP or ND entry feature for portal clients
    - Disabling traffic accounting for portal users
    - Configuring Web redirect
    - Configuring portal safe-redirect
    - Setting the maximum number of portal redirection sessions for a single user
    - Excluding an attribute from portal protocol packets
    - Configuring support of portal authentication for third-party authentication
    - Setting the user synchronization interval for portal authentication using OAuth
    - Configuring user synchronization for portal authentication using the WiFiDog protocol
    - Configuring portal authentication information report interval
    - Configuring portal advertisement push
12. (Optional.) Monitoring portal authentication
    - Enabling portal logging
    - Configuring the portal authentication monitoring feature

Prerequisites for portal authentication

The portal feature provides a solution for user identity authentication and security check. To complete user identity authentication, portal must cooperate with RADIUS.

Before you configure portal, you must complete the following tasks:

- The portal authentication server, portal Web server, and RADIUS server have been installed and configured correctly.
- To use the re-DHCP portal authentication mode, make sure the DHCP relay agent is enabled on the access device, and the DHCP server is installed and configured correctly.
- The portal client, access device, and servers can reach each other.
- To use the remote RADIUS server, configure usernames and passwords on the RADIUS server, and configure the RADIUS client on the access device. For information about RADIUS client configuration, see "Configuring AAA."
- To implement extended portal functions, install and configure IMC EAD. Make sure the ACLs configured on the access device correspond to the isolation ACL and the security ACL on the security policy server. For information about security policy server configuration on the access device, see "Configuring AAA." For installation and configuration about the security policy server, see *IMC EAD Security Policy Help*.

Configuring a remote portal authentication server

About this task

With portal authentication enabled, the device searches for a portal authentication server for a received portal request packet according to the source IP address and VPN instance information of the packet.

- If a matching portal authentication server is found, the device regards the packet valid and sends an authentication response packet to the portal authentication server. After a user logs in to the device, the user interacts with the portal authentication server as needed.
- If no matching portal authentication server is found, the device drops the packet.

Restrictions and guidelines

Do not delete a portal authentication server in use. Otherwise, users authenticated by that server cannot log out correctly.

Procedure

1. Enter system view.

   system-view

2. Create a portal authentication server and enter its view.

   portal server *server-name*

   You can create multiple portal authentication servers.

3. Specify the IP address of the portal authentication server.

   IPv4:

   ip *ipv4-address* [**vpn-instance** *vpn-instance-name*] [**key** { **cipher** | **simple** } *string*]

   IPv6:

   ipv6 *ipv6-address* [**vpn-instance** *vpn-instance-name*] [**key** { **cipher** | **simple** } *string*]

4. (Optional.) Set the destination UDP port number used by the device to send unsolicited portal packets to the portal authentication server.

   port *port-number*

   By default, the UDP port number is 50100.

   This port number must be the same as the listening port number specified on the portal authentication server.

5. (Optional.) Specify the portal authentication server type.

   server-type { **cmcc** | **imc** }

   By default, the portal authentication server type is IMC.

   The specified server type must be the same as the type of the portal authentication server actually used.

6. (Optional.) Set the maximum number of times and the interval for retransmitting a logout notification packet.

   logout-notify retry *retries* **interval** *interval*

   By default, the device does not retransmit a logout notification packet.

7. (Optional.) Configure the device to periodically register with the portal authentication server.

   server-register [**interval** *interval-value*]

   By default, the device does not register with a portal authentication server.

Configuring a portal Web server

Portal Web server tasks at a glance

To configure a portal Web server, perform the following tasks:

1. Configure basic parameters for a portal Web server
2. (Optional.) Enabling the captive-bypass feature
3. (Optional.) Configuring a match rule for URL redirection

Configure basic parameters for a portal Web server

1. Enter system view.

   `system-view`

2. Create a portal Web server and enter its view.

   `portal web-server` *`server-name`*

   You can create multiple portal Web servers.

3. Specify the VPN instance to which the portal Web server belongs.

   `vpn-instance` *`vpn-instance-name`*

   By default, the portal Web server belongs to the public network.

4. Specify the URL of the portal Web server.

   `url` *`url-string`*

   By default, no URL is specified for a portal Web server.

5. Configure the parameters to be carried in the URL when the device redirects it to users.

   `url-parameter` *`param-name`* { **`nas-id`** | **`nas-port-id`** | **`original-url`** | **`source-address`** | **`source-mac`** [**`format`** **`section`** { **`1`** | **`3`** | **`6`** } { **`lowercase`** | **`uppercase`** }] [**`encryption`** { **`aes`** | **`des`** } **`key`** { **`cipher`** | **`simple`** } *`string`*] | **`value`** *`expression`* | **`vlan`** }

   By default, no redirection URL parameters are configured.

6. (Optional.) Specify the portal Web server type.

   `server-type` { **`cmcc`** | **`imc`** | **`ise`** | **`oauth`** | **`wifidog`** }

   By default, the portal Web server type is IMC.

   This configuration is applicable to only to the remote portal service.

   The specified server type must be the same as the type of the portal Web server actually used.

Enabling the captive-bypass feature

About this task

Typically, when iOS mobile devices or some Android mobile devices are connected a portal-enabled network, the device pushes the authentication page to the mobile devices.

The captive-bypass feature enables the device to push the portal authentication page to the iOS and Android devices only when the users access the Internet by using a browser. If the users do not perform authentication but press the home button to return to the desktop, the Wi-Fi connection is terminated. To maintain the Wi-Fi connection in such cases, you can enable the optimized captive-bypass feature.

When optimized captive-bypass is enabled, the portal authentication page is automatically pushed to iOS mobile devices after they connects to the network. Users can perform authentication on the page or press the home button to return to the desktop without performing authentication, and the Wi-Fi connection is not terminated.

When an iOS client is connected to a network, it automatically sends a server reachability detection packet to determine whether the Apple server is reachable. If the server is reachable, the Wi-Fi connection displays connected. If the server is not reachable, the Wi-Fi connection is terminated.

When the network condition is poor, the device cannot detect a server reachability detection packet from an iOS mobile client within the captive-bypass detection timeout time. The client cannot receive a response for the server reachability detection packet, and therefore it determines the server to be unreachable and terminates the Wi-Fi connection. To avoid Wi-Fi disconnections caused by server reachability detection failure, set a longer captive-bypass detection timeout time when the network condition is poor.

Procedure

1. Enter system view.

   `system-view`

2. Enter portal Web server view.

   `portal web-server` *`server-name`*

3. Enable the captive-pass feature.

   `captive-bypass` [**`android`** | **`ios`** [**`optimize`**]] **`enable`**

   By default, the captive-bypass feature is disabled. The device automatically pushes the portal authentication page to iOS mobile devices and some Android mobile devices when they are connected to a portal-enabled network.

4. (Optional.) Set the captive-bypass detection timeout time.

   a. Return to system view.

      `quit`

   b. Set the captive-bypass detection timeout time.

      `portal captive-bypass optimize delay` *`seconds`*

      By default, the captive-bypass detection timeout time is 6 seconds.

Configuring a match rule for URL redirection

About this task

A URL redirection match rule matches HTTP or HTTPS requests by user-requested URL or User-Agent information, and redirects the matching HTTP or HTTPS requests to the specified redirection URL.

For a portal Web server, you can configure the **`url`** command and the **`if-match`** command for URL redirection. The **`url`** command redirects all HTTP or HTTPS requests from unauthenticated users to the portal Web server for authentication. The **`if-match`** command allows for flexible URL redirection by redirecting specific HTTP or HTTPS requests to specific redirection URLs.

Restrictions and guidelines

For a user to successfully access a redirection URL, configure a portal-free rule to allow HTTP or HTTPS requests destined for the redirection URL to pass. For information about configuring portal-free rules, see the **`portal free-rule`** command.

If both the **`url`** and **`if-match`** commands are executed, the **`if-match`** command takes priority to perform URL redirection.

If both portal safe-redirect and URL redirection match rules are configured, the device preferentially uses URL redirection match rules to perform URL redirection.

Procedure

1. Enter system view.

   `system-view`

2. Enter portal Web server view.

   `portal web-server` *`server-name`*

3. Configure a match rule for URL redirection.

   `if-match` { **`original-url`** *`url-string`* **`redirect-url`** *`url-string`* [**`url-param-encryption`** { **`aes`** | **`des`** } **`key`** { **`cipher`** | **`simple`** } *`string`*] | **`user-agent`** *`string`* **`redirect-url`** *`url-string`* }

Configuring the local portal service features

About the local portal service

After a local portal service is configured, the device acts as the portal Web server and portal authentication server to perform portal authentication on users. The portal authentication page file is saved in the root directory of the device.

Restrictions and guidelines for configuring local portal service features

For an interface to use the local portal service, the URL of the portal Web server specified for the interface must meet the following requirements:

- The IP address in the URL must be the IP address of a Layer 3 interface (except 127.0.0.1) on the device, and the IP address must be reachable to portal clients.
- The URL must be ended with **/portal/**. For example: **http://1.1.1.1/portal/**.

You must customize the authentication pages and upload them to the device.

Customizing authentication pages

About this task

Authentication pages are HTML files. Local portal authentication requires the following authentication pages:

- Logon page
- Logon success page
- Logon failure page
- Online page
- System busy page
- Logoff success page

You must customize the authentication pages, including the page elements that the authentication pages will use, for example, **back.jpg** for authentication page **Logon.htm**.

Follow the authentication page customization rules when you edit the authentication page files.

File name rules

The names of the main authentication page files are fixed (see Table 1). You can define the names of the files other than the main authentication page files. File names and directory names are case insensitive.

Table 1 Main authentication page file names

Main authentication page	File name
Logon page	logon.htm
Logon success page	logonSuccess.htm
Logon failure page	logonFail.htm
Online page Pushed after the user gets online for online notification	online.htm
System busy page Pushed when the system is busy or the user is in the logon process	busy.htm
Logoff success page	logoffSuccess.htm

Page request rules

A local portal Web service supports only Get and Post requests.

- **Get requests**—Used to get the static files in the authentication pages and allow no recursion. For example, if file **Logon.htm** includes contents that perform Get action on file **ca.htm**, file **ca.htm** cannot include any reference to file **Logon.htm**.
- **Post requests**—Used when users submit username and password pairs, log in, and log out.

Post request attribute rules

1. Observe the following requirements when editing a form of an authentication page:
   - An authentication page can have multiple forms, but there must be one and only one form whose action is **logon.cgi**. Otherwise, user information cannot be sent to the access device.
   - The username attribute is fixed as **PtUser**. The password attribute is fixed as **PtPwd**.
   - The value of the **PtButton** attribute is either **Logon** or **Logoff**, which indicates the action that the user requests.
   - A logon Post request must contain **PtUser**, **PtPwd**, and **PtButton** attributes.
   - A logoff Post request must contain the **PtButton** attribute.
2. Authentication pages **logon.htm** and **logonFail.htm** must contain the logon Post request.

   The following example shows part of the script in page **logon.htm**.

   ```
   <form action=logon.cgi method = post >
   <p>User name:<input type="text" name = "PtUser" style="width:160px;height:22px" maxlength=64>
   <p>Password :<input type="password" name = "PtPwd" style="width:160px;height:22px" maxlength=32>
   <p><input type=SUBMIT value="Logon" name = "PtButton" style="width:60px;" onclick="form.action=form.action+location.search;">
   </form>
   ```
3. Authentication pages **logonSuccess.htm** and **online.htm** must contain the logoff Post request.

   The following example shows part of the script in page **online.htm**.

   ```
   <form action=logon.cgi method = post >
   ```

```
<p><input type=SUBMIT value="Logoff" name="PtButton" style="width:60px;">
</form>
```

Page file compression and saving rules

You must compress the authentication pages and their page elements into a standard zip file.

- The name of a zip file can contain only letters, numbers, and underscores.
- The authentication pages must be placed in the root directory of the zip file.
- Zip files can be transferred to the device through FTP or TFTP and must be saved in the root directory of the device.

  Examples of zip files on the device:

  ```
  <Sysname> dir
  Directory of flash:
     1     -rw-      1405  Feb 28 2008 15:53:20   abc1.zip
     0     -rw-      1405  Feb 28 2008 15:53:31   abc2.zip
     2     -rw-      1405  Feb 28 2008 15:53:39   abc3.zip
     3     -rw-      1405  Feb 28 2008 15:53:44   abc4.zip
  2540 KB total (1319 KB free)
  ```

Redirecting authenticated users to a specific webpage

To make the device automatically redirect authenticated users to a specific webpage, do the following in logon.htm and logonSuccess.htm:

1. In logon.htm, set the target attribute of Form to **_blank**.

   See the contents in gray:

   ```
       <form method=post action=logon.cgi target="_blank">
   ```

2. Add the function for page loading pt_init() to LogonSuccess.htm.

   See the contents in gray:

   ```
       <html>
       <head>
       <title>LogonSuccess</title>
       <script type="text/javascript" language="javascript"
   src="pt_private.js"></script>
       </head>
       <body onload="pt_init();" onbeforeunload="return pt_unload();">
       ... ...
       </body>
   </html>
   ```

Configuring a local portal Web service

Prerequisites

Before you configure an HTTPS-based local portal Web service, you must complete the following tasks:

- Configure a PKI policy, obtain the CA certificate, and request a local certificate. For more information, see "Configuring PKI."
- Configure an SSL server policy, and specify the PKI domain configured in the PKI policy.

  During SSL connection establishment, the user browser might display a message that it cannot verify server identity by certificate. For users to perform portal authentication without checking such a message, configure an SSL server policy to request a client-trusted certificate on the device. The name of the policy must be **https_redirect**. For more information about SSL server policy configuration, see "Configuring SSL."

Procedure

1. Enter system view.

   system-view

2. Create an HTTP- or HTTPS-based local portal Web service and enter its view.

   portal local-web-server { **http** | **https** [**ssl-server-policy** *policy-name*] [**tcp-port** *port-number*] }

3. Specify the default authentication page file for the local portal Web service.

   default-logon-page *filename*

   The default setting varies by device model. For more information, see the command reference.

4. (Optional.) Configure the listening TCP port for the local portal Web service.

   tcp-port *port-number*

   By default, the HTTP service listening port number is 80 and the HTTPS service listening port number is the TCP port number set by the **portal local-web-server** command.

1. (Optional.) Bind an endpoint type to an authentication page file.

   logon-page bind { **device-type** { **computer** | **pad** | **phone** } | **device-name** *device-name* } * **file** *file-name*

   By default, no endpoint type is bound to an authentication page file.

2. (Optional.) Enable local portal user password modification.

   user-password modify enable

   By default, local portal user password modification is disabled.

   If global password control is enabled by using the **password-control enable network-class** command, the new password of a local portal user must meet the password control requirements. For more information about password control, see "Configuring password control."

3. Configure the redirect URL for authentication success.

   login success-url *url*-string

   By default, no redirection URL for authentication success is configured.

4. Configure the redirect URL for authentication failure.

   login failed-url *url*-string

   By default, no redirection URL for authentication failure is configured.

Configuring the User-Agent match string

About this task

When portal users use third-party software to perform portal authentication, the device checks the User-Agent string in portal authentication requests. If the User-Agent string does not include the match string specified on the device, users will fail portal authentication.

The User-Agent string includes hardware vendor, software operating system, browser, and search engine information. Perform this task to specify a string that can match the User-Agent information of the third-party software, so users can pass portal authentication by using that third-party software. For example, for users to pass portal authentication by following a WeChat official account, configure the User-Agent match string on the device as **MicroMessenger**.

Procedure

1. Enter system view.

system-view

2. Enter local portal Web service view.

   portal local-web-server { **http** | **https** [**ssl-server-policy** *policy-name*] [**tcp-port** *port-number*] }

3. Configure the User-Agent match string.

   user-agent *user-agent-string*

   By default, the User-Agent match string is **MicroMessenger**.

Enabling portal authentication on an interface

Restrictions and guidelines

When you enable portal authentication on an interface, follow these restrictions and guidelines:

- For portal authentication to take effect on an Ethernet interface, do not add the Ethernet interface to an aggregation group.
- Cross-subnet authentication mode (**layer3**) does not require Layer 3 forwarding devices between the access device and the portal authentication clients. However, if a Layer 3 forwarding device exists between the authentication client and the access device, you must use the cross-subnet portal authentication mode.
- You can enable both IPv4 portal authentication and IPv6 portal authentication on an interface.

When you configure re-DHCP portal authentication on an interface, follow these restrictions and guidelines:

- Make sure the interface has a valid IP address before you enable re-DHCP portal authentication on the interface.
- With re-DHCP portal authentication, configure authorized ARP on the interface as a best practice to make sure only valid users can access the network. With authorized ARP configured on the interface, the interface learns ARP entries only from the users who have obtained a public address from DHCP.
- For successful re-DHCP portal authentication, make sure the BAS-IP or BAS-IPv6 attribute value is the same as the device IP address specified on the portal authentication server. To configure the attribute, use the **portal** { **bas-ip** | **bas-ipv6** } command.
- An IPv6 portal server does not support re-DHCP portal authentication.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable portal authentication.

   IPv4:

   portal enable method { **direct** | **layer3** | **redhcp** }

   IPv6:

   portal ipv6 enable method { **direct** | **layer3** }

   By default, portal authentication is disabled.

Specifying a portal Web server on an interface

About this task

With a portal Web server specified on an interface, the device redirects the HTTP requests of portal users on the interface to the portal Web server.

You can specify both an IPv4 portal Web server and an IPv6 portal Web server on an interface.

Procedure

1. Enter system view.
   `system-view`
2. Enter Layer 3 interface view.
   `interface` *`interface-type interface-number`*
3. Specify a portal Web server on the interface.
   `portal` [**`ipv6`**] **`apply web-server`** *`server-name`* [**`secondary`**]
   By default, no portal Web servers are specified on an interface.

Configuring a portal preauthentication domain

About this task

A portal preauthentication domain defines user attributes assigned to preauthentication portal users on a portal-enabled interface after the users obtain IP addresses. Before the preauthentication users pass portal authentication, they have limited access to the network based on the assigned user attributes (such as ACL and CAR). After the users pass portal authentication, they are assigned new attributes by the AAA server. After the users go offline, they are re-assigned user attributes in the preauthentication domain.

Restrictions and guidelines

The portal preauthentication domain takes effect only on portal users with IP addresses obtained through DHCP or DHCPv6.

The portal preauthentication domain does not take effect on interfaces enabled with cross-subnet portal authentication.

Make sure you specify an existing ISP domain as a portal preauthentication domain. If the specified ISP domain does not exist, the device might operate incorrectly.

You must delete a preauthentication domain (by using the **`undo portal`** [**`ipv6`**] **`pre-auth domain`** command) and reconfigure it in the following situations:

- You create the ISP domain after specifying it as the preauthentication domain.
- You delete the specified ISP domain and then re-create it.

For the authorization ACL in the preauthentication domain, the following rules apply:

- If the traffic of preauthentication users matches a rule in the ACL, the device processes the traffic based on the permit or deny statement of the rule.
- If the ACL does not exist or the destination address permitted by a rule in the ACL is set to **`any`**, the device does not control user access. Users can access any network resources without passing portal authentication.
- If the ACL does not have any rules, the device allows users to access network resources only after the users pass authentication.

- If the traffic of preauthentication users does not match any rule in the ACL, the device pushes the authentication page to the users. The users can access the network resources after passing authentication.
- If the ACL contains rules that specify a source address, users might not be able to get online. Do not specify a source IPv4, IPv6, or MAC address when you configure a rule in the ACL.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Specify a portal preauthentication domain.

   `portal` `[` **`ipv6`** `]` **`pre-auth domain`** *`domain-name`*

   By default, no portal preauthentication domain is specified.

Specifying a preauthentication IP address pool

About this task

You must specify a preauthentication IP address pool on a portal-enabled Layer 3 interface in the following situation:

- Portal users access the network through a subinterface of the portal-enabled Layer 3 interface.
- The subinterface does not have an IP address.
- Portal users need to obtain IP addresses through DHCP.

After a user connects to a portal-enabled interface, the user uses an IP address for portal authentication according to the following rules:

- If the interface is configured with a preauthentication IP address pool, the user uses the following IP address:
  - If the client is configured to obtain an IP address automatically through DHCP, the user obtains an address from the specified IP address pool.
  - If the client is configured with a static IP address, the user uses the static IP address. However, if the interface does not have an IP address, users using static IP addresses cannot pass authentication.
- If the interface has an IP address but no preauthentication IP pool specified, the user uses the static IP address or the IP address obtained from a DHCP server.
- If the interface has no IP address or preauthentication IP pool specified, the user cannot perform portal authentication.

After the user passes portal authentication, the AAA server authorizes an IP address pool for re-assigning an IP address to the user. If no authorized IP address pool is deployed, the user continues using the previous IP address.

Restrictions and guidelines

This configuration takes effect only when the direct IPv4 portal authentication is enabled on the interface.

Make sure the specified IP address pool exists and is complete. Otherwise, the user cannot obtain the IP address and cannot perform portal authentication.

If the portal user does not perform authentication or fails to pass authentication, the assigned IP address is still retained.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Specify a preauthentication IP address pool on the interface.

   `portal` [**`ipv6`**] **`pre-auth ip-pool`** *`pool-name`*

   By default, no preauthentication IP address pool is specified on an interface.

Specifying a portal authentication domain

About portal authentication domains

An authentication domain defines a set of authentication, authorization, and accounting policies. Each portal user belongs to an authentication domain and is authenticated, authorized, and accounted in the domain.

With an authentication domain specified on an interface, the device uses the authentication domain for AAA of portal users. This allows for flexible portal access control.

Restrictions and guidelines for specifying a portal authentication domain

The device selects the authentication domain for a portal user in this order:

1. ISP domain specified for the interface.
2. ISP domain carried in the username.
3. System default ISP domain.

If the chosen domain does not exist on the device, the device searches for the ISP domain configured to accommodate users assigned to nonexistent domains. If no such ISP domain is configured, user authentication fails. For information about ISP domains, see "Configuring AAA."

If an authorization VPN instance is specified in the authentication domain, follow these restrictions and guidelines:

- Make sure the authorization VPN instance exists. If you specify a nonexistent VPN instance, users cannot come online.
- When users are online, do not delete the authorization VPN instance. Deleting the authorization VPN instance will log out the users.

For the authorization ACL in the authentication domain, the following rules apply:

- If the user traffic matches a rule in the ACL, the device processes the traffic based on the permit or deny statement of the rule.
- If the user traffic does not match any rule in the ACL, the device permits the traffic. To deny such traffic, configure the last rule in the ACL to deny all packets by using the **`rule deny ip`** command.
- If the ACL contains rules that specify a source address, users might not be able to get online. Do not specify a source IPv4, IPv6, or MAC address when you configure a rule in the ACL.

Specifying a portal authentication domain on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Specify an portal authentication domain on the interface.

   portal [**ipv6**] **domain** *domain-name*

   By default, no portal authentication domain is specified on an interface.

   You can specify both an IPv4 portal authentication domain and an IPv6 portal authentication domain on an interface.

Controlling portal user access

Configuring a portal-free rule

About this task

A portal-free rule allows specified users to access specified external websites without portal authentication.

The matching items for a portal-free rule include the host name, source/destination IP address, TCP/UDP port number, source MAC address, access interface, and VLAN. Packets matching a portal-free rule will not trigger portal authentication, so users sending the packets can directly access the specified external websites.

Restrictions and guidelines for configuring a portal-free rule

If you specify both a VLAN and an interface, the interface must belong to the VLAN. If the interface does not belong to the VLAN, the portal-free rule does not take effect.

You cannot configure two or more portal-free rules with the same filtering criteria. Otherwise, the system prompts that the rule already exists.

Regardless of whether portal authentication is enabled or not, you can only add or remove a portal-free rule. You cannot modify it.

The ACL used by a portal-free rule can contain only IP address object groups and IP quintuples (source and destination IP addresses, source and destination port numbers, and transport layer protocol).

Configuring an IP-based portal-free rule

1. Enter system view.

   system-view

2. Configure an IP-based portal-free rule.

   IPv4:

   portal free-rule *rule-number* { **destination ip** { *ipv4-address* { *mask-length* | *mask* } | **any** } [**tcp** *tcp-port-numbe*r | **udp** *udp-port-number*] | **source ip** { *ipv4-address* { *mask-length* | *mask* } | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] } * [**interface** *interface-type interface-number*]

   IPv6:

   portal free-rule *rule-number* { **destination ipv6** { *ipv6-address prefix-length* | **any** } [**tcp** *tcp-port-numbe*r | **udp** *udp-port-number*] |

source ipv6 { *ipv6-address prefix-length* | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] } * [**interface** *interface-type interface-number*]

By default, no IP-based portal-free rules are configured.

Configuring a source-based portal-free rule

1. Enter system view.

   system-view

2. Configure a source-based portal-free rule.

   portal free-rule *rule-number* **source** { **interface** *interface-type interface-number* | **mac** *mac-address* | **object-group** *object-group-name* | **vlan** *vlan-id* } * }

   By default, no source-based portal-free rules are configured.

   The **vlan** *vlan-id* option takes effect only on portal users that access the network through VLAN interfaces.

Configuring a destination-based portal-free rule

1. Enter system view.

   system-view

2. Configure a destination-based portal-free rule.

   portal free-rule *rule-number* **destination** *host-name*

   By default, no destination-based portal-free rules are configured.

   Before you configure destination-based portal-free rules, make sure a DNS server has been deployed in the network.

Configuring an ACL-based portal-free rule

1. Enter system view.

   system-view

2. Configure an ACL-based portal-free rule.

   IPv4:

   portal free-rule *rule-number* **acl** *acl-number*

   IPv6:

   portal free-rule *rule-number* **acl ipv6** *acl-number*

Configuring a description for a portal-free rule

1. Enter system view.

   system-view

2. Configure a description for a portal-free rule.

   portal free-rule *rule-number* **description** *text*

   By default, no description is configured for a portal-free rule.

Configuring an authentication source subnet

About this task

By configuring authentication source subnets, you specify that only HTTP or HTTPS packets from users on the authentication source subnets can trigger portal authentication. If an unauthenticated user is not on any authentication source subnet, the access device discards all the user's HTTP or HTTPS packets that do not match any portal-free rule.

Restrictions and guidelines

Authentication source subnets apply only to cross-subnet portal authentication.

In direct or re-DHCP portal authentication mode, a portal user and its access interface (portal-enabled) are on the same subnet. It is not necessary to specify the subnet as the authentication source subnet.

- In direct mode, the access device regards the authentication source subnet as any source IP address.
- In re-DHCP mode, the access device regards the authentication source subnet on an interface as the subnet to which the private IP address of the interface belongs.

If both authentication source subnets and destination subnets are configured on an interface, only the authentication destination subnets take effect.

You can configure multiple authentication source subnets. If the source subnets overlap, the subnet with the largest address scope (with the smallest mask or prefix) takes effect.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Configure a portal authentication source subnet.

   IPv4:

   `portal layer3 source` *`ipv4-network-address`* { *`mask-length`* | *`mask`* }

   By default, users from any subnets must pass portal authentication.

   IPv6:

   `portal ipv6 layer3 source` *`ipv6-network-address prefix-length`*

   By default, users from any subnets must pass portal authentication.

Configuring an authentication destination subnet

About this task

By configuring authentication destination subnets, you specify that users trigger portal authentication only when they accessing the specified subnets (excluding the destination IP addresses and subnets specified in portal-free rules). Users can access other subnets without portal authentication.

Restrictions and guidelines

If both authentication source subnets and destination subnets are configured on an interface, only the authentication destination subnets take effect.

You can configure multiple authentication destination subnets. If the destination subnets overlap, the subnet with the largest address scope (with the smallest mask or prefix) takes effect.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Configure a portal authentication destination subnet.

   IPv4:

portal free-all except destination *ipv4-network-address* { *mask-length* | *mask* }

IPv6:

portal ipv6 free-all except destination *ipv6-network-address prefix-length*

By default, users accessing any subnets must pass portal authentication.

Configuring a portal-forbidden rule

About this task

Portal-forbidden rules are used to filter user packets from the specified sources or destined for the specified destinations. The device drops HTTP or HTTPS packets that match the portal-forbidden rules.

Restrictions and guidelines

In a portal-forbidden rule, the source and destination IP addresses must be of the same IP type, and the source and destination ports must be of the same transport protocol type.

You can configure multiple portal-forbidden rules.

If the source or destination information in a portal-free rule and that in a portal-forbidden rule overlap, the portal-forbidden rule takes effect.

If you specify a destination host name in a portal-forbidden rule, the device drops users' DNS query packets for the specified host name. In addition, if a DNS server is correctly configured on the device, the device also drops user packets destined for the IP address resolved from the specified host name. If the DNS server is not correctly configured, the rule does not take effect on user packets destined for that IP address.

Procedure

1. Enter system view.

   system-view

2. Configure a portal-forbidden rule.

   IPv4:

portal forbidden-rule *rule-number* [**source** { **ip** { *ipv4-address* { *mask-length* | *mask* } | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] | **ssid** *ssid-name* } *] **destination** { *host-name* | **ip** { *ipv4-address* { *mask-length* | *mask* } | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] }

   IPv6:

   portal forbidden-rule *rule-number* [**source** { **ipv6** { *ipv6-address prefix-length* | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] | **ssid** *ssid-name* } *] **destination** { *host-name* | **ipv6** { *ipv6-address prefix-length* | **any** } [**tcp** *tcp-port-number* | **udp** *udp-port-number*] }

   By default, portal-forbidden rules are configured.

Specifying port numbers of Web proxy servers

About this task

To allow HTTP or HTTPS requests proxied by Web proxy servers to trigger portal authentication, specify the port numbers of the Web proxy servers on the device. If a Web proxy server port is not specified on the device, HTTP or HTTPS requests proxied by the Web proxy server are dropped, and portal authentication cannot be triggered.

Restrictions and guidelines

Do not specify TCP port number 80 or 443 as the port numbers for Web proxy servers because 80 and 443 are port numbers reserved for portal.

You can specify a maximum of 64 Web proxy server ports for HTTP and HTTPS.

Do not specify the same Web proxy server port for HTTP and HTTPS.

If a user's browser uses the Web Proxy Auto-Discovery (WPAD) protocol to discover Web proxy servers, you must perform the following tasks on the device:

- Specify the port numbers of the Web proxy servers on the device.
- Configure portal-free rules to allow user packets destined for the IP address of the WPAD server to pass without authentication.

If portal users enable Web proxy in their browsers, the users must add the IP address of the portal authentication server as a proxy exception in their browsers. Then, HTTP or HTTPS packets that the users send to the portal authentication server will not be sent to Web proxy servers.

Procedure

1. Enter system view.

   system-view

2. Specify the port number of a Web proxy server.

   portal web-proxy { **http** | **https** } **port** *port-number*

   By default, no port numbers of Web proxy servers are specified. Proxied HTTP and HTTPS requests are dropped.

Setting the maximum number of portal users

About this task

Perform this task to control the total number of portal users in the system, and the maximum number of IPv4 or IPv6 portal users on an interface.

Restrictions and guidelines for setting the maximum number of portal users

Make sure the maximum combined number of IPv4 and IPv6 portal users specified on all interfaces does not exceed the system-allowed maximum number. Otherwise, the exceeding number of portal users will not be able to log in to the device.

Setting the global maximum number of portal users

1. Enter system view.

   system-view

2. Set the global maximum number of portal users.

   portal max-user *max-number*

   By default, no limit is placed on the number of portal users allowed in the system.

   If you set the global maximum number smaller than the number of current online portal users on the device, this configuration still takes effect. The online users are not affected but the system forbids new portal users to log in.

Setting the maximum number of portal users on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Set the maximum number of portal users.

portal { **ipv4-max-user** | **ipv6-max-user** } *max-number*

By default, no limit is placed on the number of portal users on an interface.

If you set the maximum number smaller than the current number of portal users on an interface, this configuration still takes effect. The online users are not affected but the system forbids new portal users to log in from the interface.

Enabling strict-checking on portal authorization information

About this task

The strict checking feature allows a portal user to stay online only when the authorization information for the user is successfully deployed.

Enabling strict checking on portal authentication information on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable strict checking on portal authorization information.

   portal authorization acl strict-checking

   By default, strict checking on portal authorization information is disabled on an interface. Portal users stay online even when the authorized ACL or user profile does not exist or the device fails to deploy the authorized ACL or user profile

△ **CAUTION:**

- The device logs out a portal user if the authorized ACL does not exist on the device or the device fails to deploy the authorized ACL.

Allowing only users with DHCP-assigned IP addresses to pass portal authentication

About this task

This feature allows only users with DHCP-assigned IP addresses to pass portal authentication. Use this feature to ensure that only users with valid IP addresses can access the network.

Restrictions and guidelines

This feature takes effect only when the device acts as both the access device and the DHCP server.

Configuration of this feature does not affect the online portal users.

Allowing only users with DHCP-assigned IP addresses to pass portal authentication on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Allow only users with DHCP-assigned IP addresses to pass portal authentication.

   portal [**ipv6**] **user-dhcp-only**

   By default, both users with IP addresses obtained through DHCP and users with static IP addresses can pass authentication to come online.

△ **CAUTION:**

- With this feature enabled, users with static IP addresses cannot pass portal authentication to come online.
- To ensure that IPv6 users can pass portal authentication when this feature is enabled, disable the temporary IPv6 address feature on terminal devices. Otherwise, IPv6 users will use temporary IPv6 addresses to access the IPv6 network and will fail portal authentication.

Blocking portal users that fail portal authentication

About this task

This feature prevents exhaustive password cracking. It blocks a portal user if the user consecutively fails authentication for the specified times within the failure detection period. All authentication requests from the user are dropped by the device till the blocking times out. The blocked portal user can perform portal authentication again when the blocking timeout time expires.

Restrictions and guidelines

This feature does not block preauthentication portal users.

Procedure

1. Enter system view.

   `system-view`

2. Configure the device to block portal users that fail portal authentication for the specified times within the specified period.

   `portal user-block failed-times` *`failed-times`* **`period`** *`period`* [**`method`** { **`ip`** | **`mac`** | **`username`** }]

   By default, the device does not block portal users that fail portal authentication.

3. Set the portal user blocking timeout time.

   `portal user-block reactive` *`period`*

   By default, the portal user blocking timeout time is 30 minutes.

   If you set the portal user blocking timeout time to 0 minutes, blocked portal users cannot perform portal authentication any more.

Configuring support of portal authentication for dual stack

About this task

Typically, IPv4 portal users can access only the IPv4 network after passing portal authentication, and IPv6 portal users can access only the IPv6 network after passing portal authentication. To allow portal users to access both IPv4 and IPv6 networks after passing one type (IPv4 or IPv6) of portal authentication, enable the portal dual-stack feature.

Configuring support of portal authentication for dual stack on an interface

1. Enter system view.

   `system-view`

2. Enable separate IPv4 and IPv6 traffic statistics in portal user offline logs.

   `portal user-log traffic-separate`

   By default, IPv4 and IPv6 traffic statistics of a portal user are collectively counted as IPv4 traffic statistics in portal user offline logs.

3. Enter Layer 3 interface view.

interface *interface-type interface-number*

4. Enable the portal dual-stack feature on the interface.

   portal dual-stack enable

   By default, the portal dual-stack feature is disabled on an interface.

5. Enable separate IPv4 and IPv6 traffic statistics for dual-stack portal users on the interface.

   portal dual-stack traffic-separate enable

   By default, separate IPv4 and IPv6 traffic statistics is disabled for dual-stack portal users on an interface. The device collects IPv4 and IPv6 traffic statistics collectively.

6. Enable the dual IP feature to carry both IPv4 and IPv6 addresses for single-stack users in remote portal authentication.

   portal dual-ip enable

   By default, the dual IP feature is disabled.

Enabling intra-VLAN roaming for portal users

About this task

If intra-VLAN roaming is enabled for portal users on a VLAN interface, an online portal user can access resources from any Layer 2 port in the VLAN without re-authentication.

If intra-VLAN roaming is disabled for portal users, to access external network resources from a Layer 2 port different from the current access port in the VLAN, the user must do the following:

1. Logs out from the current port.
2. Re-authenticates on the new Layer 2 port.

Restrictions and guidelines

Intra-VLAN roaming takes effect only on portal users logging in from VLAN interfaces. It does not take effect on portal users logging in from common Layer 3 interface.

You cannot enable intra-VLAN roaming when online portal users or preauthentication portal users exist on the device.

For intra-VLAN roaming to take effect, you must disable the Rule ARP or ND entry feature by using the **undo portal refresh** { **arp** | **nd** } **enable** command.

Procedure

1. Enter system view.

   system-view

2. Enable intra-VLAN roaming for portal users.

   portal roaming enable

   By default, intra-VLAN roaming is disabled for portal users.

Enabling outgoing packets filtering

About this task

When you enable this feature on a portal-enabled interface, the device permits the interface to send the following packets:

- Packets whose destination IP addresses are IP addresses of authenticated portal users.
- Packets that match portal-free rules.

Other outgoing packets on the interface are dropped.

Enabling outgoing packets filtering on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable outgoing packets filtering on the interface.

   portal [**ipv6**] **outbound-filter enable**

   By default, outgoing packets filtering is disabled on an interface. The interface can send any packets.

Configuring the portal fail-permit feature

About this task

You can configure the portal fail-permit feature on an interface. When the access device detects that the portal authentication server or portal Web server is unreachable, it allows users to have network access without portal authentication.

If you enable fail-permit for both the portal authentication server and the portal Web servers, the device does the following:

- Disables portal authentication when the portal authentication server is unreachable or all the portal Web servers are unreachable.
- Resumes portal authentication when both the portal authentication and Web servers are reachable.

After portal authentication resumes, unauthenticated users must pass portal authentication to access the network. Users who have passed portal authentication before the fail-permit event can continue accessing the network.

Configuring portal fail-permit on an interface

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable portal fail-permit for a portal authentication server on the interface.

   portal [**ipv6**] **fail-permit server** *server-name*

   By default, portal fail-permit is disabled for a portal authentication server on an interface.

4. Enable portal fail-permit for portal Web servers on the interface.

   portal [**ipv6**] **fail-permit web-server**

   By default, portal fail-permit is disabled for portal Web servers on an interface.

Configuring portal detection features

Configuring online detection of portal users

About this task

Use the online detection feature to quickly detect abnormal logouts of portal users. Configure ARP or ICMP detection for IPv4 portal users. Configure ND or ICMPv6 detection for IPv6 portal users.

If the device receives no packets from a portal user within the idle time, the device detects the user's online status as follows:

- **ICMP or ICMPv6 detection**—Sends ICMP or ICMPv6 requests to the user at configurable intervals to detect the user status.
  - If the device receives a reply within the maximum number of detection attempts, it considers that the user is online and stops sending detection packets. Then the device resets the idle timer and repeats the detection process when the timer expires.
  - If the device receives no reply after the maximum number of detection attempts, the device logs out the user.
- **ARP or ND detection**—Sends ARP or ND requests to the user and detects the ARP or ND entry status of the user at configurable intervals.
  - If the ARP or ND entry of the user is refreshed within the maximum number of detection attempts, the device considers that the user is online and stops detection. Then the device resets the idle timer and repeats the detection process when the timer expires.
  - If the ARP or ND entry of the user is not refreshed after the maximum number of detection attempts, the device logs out the user.

Restrictions and guidelines

ARP detection and ND detection apply only to direct and re-DHCP portal authentication. ICMP detection applies to all portal authentication modes.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Configure online detection of portal users.

   IPv4:

   portal user-detect type { **arp** | **icmp** } [**retry** *retries*] [**interval** *interval*] [**idle** *time*]

   IPv6:

   portal ipv6 user-detect type { **icmpv6** | **nd** } [**retry** *retries*] [**interval** *interval*] [**idle** *time*]

   By default, online detection is disabled for portal users on an interface.

Configuring portal authentication server detection

About this task

During portal authentication, if the communication between the access device and portal authentication server is broken, new portal users are not able to log in. Online portal users are not able to log out normally.

To address this problem, the access device needs to be able to detect the reachability changes of the portal server quickly and take corresponding actions to deal with the changes.

The portal authentication server detection feature enables the device to periodically detect portal packets sent by a portal authentication server to determine the reachability of the server. If the device receives a portal packet within a detection timeout (**timeout** *timeout*) and the portal packet is valid, the device considers the portal authentication server to be reachable. Otherwise, the device considers the portal authentication server to be unreachable.

Portal packets include user login packets, user logout packets, and heartbeat packets. Heartbeat packets are periodically sent by a server. By detecting heartbeat packets, the device can detect the server's actual status more quickly than by detecting other portal packets.

Restrictions and guidelines

The portal authentication server detection feature takes effect only when the device has a portal-enabled interface.

Only the IMC portal authentication server supports sending heartbeat packets. To test server reachability by detecting heartbeat packets, you must enable the server heartbeat feature on the IMC portal authentication server.

You can configure the device to take one or more of the following actions when the server reachability status changes:

- Sending a trap message to the NMS. The trap message contains the name and current state of the portal authentication server.
- Sending a log message, which contains the name, the current state, and the original state of the portal authentication server.
- Enabling portal fail-permit. When the portal authentication server is unreachable, the portal fail-permit feature on an interface allows users on the interface to have network access. When the server recovers, it resumes portal authentication on the interface. For more information, see "Configuring the portal fail-permit feature."

Make sure the detection timeout configured on the device is greater than the server heartbeat interval configured on the portal authentication server.

Procedure

1. Enter system view.

   system-view

2. Enter portal authentication server view.

   portal server *server-name*

3. Configure portal authentication server detection.

   server-detect [**timeout** *timeout*] { **log** | **trap** } *

   By default, portal authentication server detection is disabled.

Configuring portal Web server detection

About this task

A portal authentication process cannot complete if the communication between the access device and the portal Web server is broken. To address this problem, you can enable portal Web server detection on the access device.

With the portal Web server detection feature, the access device simulates a Web access process to initiate a TCP connection to the portal Web server. If the TCP connection can be established successfully, the access device considers the detection successful, and the portal Web server is reachable. Otherwise, it considers the detection to have failed.

You can configure the following detection parameters:

- **Detection interval**—Interval at which the device detects the server reachability.
- **Maximum number of consecutive failures**—If the number of consecutive detection failures reaches this value, the access device considers that the portal Web server is unreachable.

You can configure the device to take one or more of the following actions when the server reachability status changes:

- Sending a trap message to the NMS. The trap message contains the name and current state of the portal Web server.
- Sending a log message, which contains the name, the current state, and the original state of the portal Web server.
- Enabling portal fail-permit. When the portal Web server is unreachable, the portal fail-permit feature on an interface allows users on the interface to have network access. When the server recovers, it resumes portal authentication on the interface. For more information, see "Configuring the portal fail-permit feature."

Restrictions and guidelines

The portal Web server detection feature takes effect only when the URL of the portal Web server is specified and the device has a portal-enabled interface.

Procedure

1. Enter system view.

   system-view

2. Enter portal Web server view.

   portal web-server *server-name*

3. Configure portal Web server detection.

   server-detect [**interval** *interval*] [**retry** *retries*] { **log** | **trap** } *

   By default, portal Web server detection is disabled.

4. Configure the URL and the type for portal Web server detection.

   server-detect url *string* [**detect-type** { **http** | **tcp** }]

   By default, the URL for portal Web server detection is the URL of the portal Web server. The type of portal Web server detection is TCP detection.

Enabling DHCP packet capture

About this task

This feature enables the AC to detect the online status of portal users by capturing DHCP packets of the portal users.

When this feature is enabled, the AC captures DHCP packets between a portal user and the DHCP server and obtains the IP address lease information of the user. The AC then detects the online status of the portal user as follows:

- If the AC captures a DHCP lease renewal packet from the portal user before the lease expires, the AC determines that the portal user is online.
- If no DHCP lease renewal packet is captured before the lease expires, the AC forcibly logs out the portal user.

For more information about DHCP packets, see DHCP configuration in *Layer 3—IP Services Configuration Guide*.

The timeout time of the DHCP packet capture timer is the same as the IP address lease time in DHCP packets. This timer resets each time a DHCP packet is captured.

Procedure

1. Enter system view.

   system-view

2. Enable DHCP packet capture to detect online status of portal users.

   portal idle-cut dhcp-capture enable

   By default, DHCP packet capture is disabled.

Configuring portal user synchronization

About this task

Once the access device loses communication with a portal authentication server, the portal user information on the access device and that on the portal authentication server might be inconsistent after the communication resumes. To address this problem, the device provides the portal user synchronization feature. This feature is implemented by sending and detecting portal synchronization packets, as follows:

1. The portal authentication server sends the online user information to the access device in a synchronization packet at the user heartbeat interval.

   The user heartbeat interval is set on the portal authentication server.

2. Upon receiving the synchronization packet, the access device compares the users carried in the packet with its own user list and performs the following operations:
   - If a user contained in the packet does not exist on the access device, the access device informs the portal authentication server to delete the user. The access device starts the synchronization detection timer (**`timeout`** *`timeout`*) immediately when a user logs in.
   - If the user does not appear in any synchronization packet within a synchronization detection interval, the access device considers the user does not exist on the portal authentication server and logs the user out.

Restrictions and guidelines

Portal user synchronization requires a portal authentication server to support the portal user heartbeat function. Only the IMC portal authentication server supports the portal user heartbeat function. To implement the portal user synchronization feature, you also need to configure the user heartbeat function on the portal authentication server. Make sure the user heartbeat interval configured on the portal authentication server is not greater than the synchronization detection timeout configured on the access device.

Deleting a portal authentication server on the access device also deletes the user synchronization configuration for the portal authentication server.

Procedure

1. Enter system view.

   `system-view`

2. Enter portal authentication server view.

   `portal server` *`server-name`*

3. Configure portal user synchronization.

   `user-sync timeout` *`timeout`*

   By default, portal user synchronization is disabled.

Configuring portal packet attributes

Configuring the BAS-IP or BAS-IPv6 attribute

About this task

If the device runs Portal 2.0, the unsolicited packets sent to the portal authentication server must carry the BAS-IP attribute. If the device runs Portal 3.0, the unsolicited packets sent to the portal authentication server must carry the BAS-IP or BAS-IPv6 attribute.

After this attribute is configured, the source IP address for unsolicited notification portal packets the device sends to the portal authentication server is the configured BAS-IP or BAS-IPv6 address. If the attribute is not configured, the source IP address of the portal packets is the IP address of the packet output interface.

Restrictions and guidelines

During a re-DHCP portal authentication or mandatory user logout process, the device sends portal notification packets to the portal authentication server. For the authentication or logout process to complete, make sure the BAS-IP or BAS-IPv6 attribute is the same as the device IP address specified on the portal authentication server.

You must configure the BAS-IP or BAS-IPv6 attribute on a portal authentication-enabled interface if the following conditions are met:

- The portal authentication server is an IMC server.
- The portal device IP address specified on the portal authentication server is not the IP address of the portal packet output interface.

Configuring the BAS-IP or BAS-IPv6 attribute on an interface

1. Enter system view.
   `system-view`
2. Enter Layer 3 interface view.
   `interface` *`interface-type interface-number`*
3. Configure the BAS-IP or BAS-IPv6 attribute.
   IPv4:
   `portal bas-ip` *`ipv4-address`*
   By default, the BAS-IP attribute of an IPv4 portal reply packet is the source IPv4 address of the packet. The BAS-IP attribute of an IPv4 portal notification packet is the IPv4 address of the packet's output interface.
   IPv6:
   `portal bas-ipv6` *`ipv6-address`*
   By default, the BAS-IPv6 attribute of an IPv6 portal reply packet is the source IPv6 address of the packet. The BAS-IPv6 attribute of an IPv6 portal notification packet is the IPv6 address of the packet's output interface.

Specifying the device ID

About this task

The portal authentication server uses device IDs to identify the devices that send protocol packets to the portal server.

Restrictions and guidelines

Make sure the configured device ID is different than any other access devices communicating with the same portal authentication server.

Procedure

1. Enter system view.
   `system-view`
2. Specify the device ID.
   `portal device-id` *`device-id`*
   By default, a device is not configured with a device ID.

Configuring attributes for RADIUS packets

Specifying a format for the NAS-Port-Id attribute

About this task

RADIUS servers from different vendors might require different formats of the NAS-Port-Id attribute in the RADIUS packets. You can specify the NAS-Port-Id attribute format as required.

The device supports predefined formats (format 1, 2, 3, and 4). For more information about the formats, see portal commands in *Security Command Reference*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify the format for the NAS-Port-Id attribute.

   ```
   portal nas-port-id format { 1 | 2 | 3 | 4 }
   ```

   By default, the format for the NAS-Port-Id attribute is format 2.

Applying a NAS-ID profile to an interface

About this task

By default, the device sends its device name in the NAS-Identifier attribute of all RADIUS requests.

A NAS-ID profile enables you to send different NAS-Identifier attribute strings in RADIUS requests from different VLANs. The strings can be organization names, service names, or any user categorization criteria, depending on the administrative requirements.

For example, map the NAS-ID **companyA** to all VLANs of company A. The device will send **companyA** in the NAS-Identifier attribute for the RADIUS server to identify requests from any Company A users.

Restrictions and guidelines

You can apply a NAS-ID profile to a portal-enabled interface. If no NAS-ID profile is specified on the interface or no matching NAS-ID is found in the specified profile, the device uses the device name as the interface NAS-ID.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Create a NAS-ID profile and enter NAS-ID profile view.

   ```
   aaa nas-id profile profile-name
   ```

   For more information about this command, see AAA commands in *Security Command Reference*.

3. Configure a NAS ID and VLAN binding in the profile.

   ```
   nas-id nas-identifier bind vlan vlan-id
   ```

   For more information about this command, see AAA commands in *Security Command Reference*.

4. Specify the NAS-ID profile on the interface.

   a. Return to system view.

      ```
      quit
      ```

b. Enter Layer 3 interface view.

   interface *interface-type interface-number*

c. Specify the NAS-ID profile on the interface.

   portal nas-id-profile *profile-name*

Configuring MAC-based quick portal authentication

Restrictions and guidelines for configuring MAC-based quick portal authentication

Only IPv4 direct authentication supports MAC-based quick portal authentication.

For MAC-based quick portal authentication to take effect on an interface configured with a portal preauthentication domain, set the free-traffic threshold to 0 bytes.

Configuring a remote MAC binding server

About this task

You can configure multiple MAC binding servers on the device. For each server, you can configure MAC binding server parameters, such as the server's IP address, VPN instance, port number, and the free-traffic threshold.

Procedure

1. Enter system view.

   system-view

2. Create a MAC binding server and enter its view.

   portal mac-trigger-server *server-name*

3. Configure a MAC binding server.

   - Specify the IP address of the MAC binding server.

     ip *ipv4-address* [**vpn-instance** *vpn-instance-name*] [**key** { **cipher** | **simple** } *string*]

     By default, no IP address is specified for a MAC binding server.

   - (Optional.) Set the UDP port number on which the MAC binding server listens for MAC binding query packets.

     port *port-number*

     By default, the MAC binding server listens for MAC binding query packets on UDP port 50100.

   - (Optional.) Set the maximum number of attempts and the interval for sending MAC binding queries to the MAC binding server.

     binding-retry { *retries* | **interval** *interval* } *

     By default, the maximum number of query attempts is 3 and the query interval is 1 second.

   - (Optional.) Specify the type of the MAC binding server.

     server-type { **cmcc** | **imc** }

     By default, the type of a MAC binding server is IMC.

4. (Optional.) Set the free-traffic threshold.

free-traffic threshold *value*

By default, the free-traffic threshold is 0 bytes.

5. (Optional.) Set the NAS-Port-Type value carried in RADIUS requests sent to the RADIUS server.

   nas-port-type *value*

   By default, the NAS-Port-Type value carried in RADIUS requests is 15.

6. (Optional.) Specify the version of the portal protocol.

   version *version-number*

   By default, the version of the portal protocol is 1.

7. (Optional.) Set the timeout the device waits for portal authentication to complete after receiving the MAC binding query response.

   authentication-timeout *minutes*

   By default, the portal authentication timeout time is 3 minutes.

8. (Optional.) Set the aging time for MAC-trigger entries.

   aging-time *seconds*

   By default, the aging time for MAC-trigger entries is 300 seconds.

9. (Optional.) Enable AAA failure unbinding.

   aaa-fail nobinding enable

   By default, AAA failure unbinding is disabled.

Configuring a local MAC binding server

About this task

To provide MAC-based quick portal authentication for local portal users, perform this task so that the access device acts as the local MAC binding server.

You can configure multiple MAC binding servers on the device. For each server, you can configure parameters related to MAC-based quick portal authentication.

Procedure

1. Enter system view.

   system-view

2. Create a MAC binding server and enter its view.

   portal mac-trigger-server *server-name*

3. Enable local MAC-based quick portal authentication.

   local-binding enable

   By default, local MAC-based quick portal authentication is disabled.

4. (Optional.) Set the free-traffic threshold.

   free-traffic threshold *value*

   By default, the free-traffic threshold is 0 bytes.

5. (Optional.) Set the aging time for local MAC-account binding entries.

   local-binding aging-time *minutes*

   By default, the aging time for local MAC-account binding entries is 720 minutes.

6. (Optional.) Set the aging time for MAC-trigger entries.

   aging-time *seconds*

   By default, the aging time for MAC-trigger entries is 300 seconds.

7. (Optional.) Enable AAA failure unbinding.

   `aaa-fail nobinding enable`

   By default, AAA failure unbinding is disabled.

Specifying a MAC binding server on an interface

About this task

After a MAC binding server is specified on an interface, the device can implement MAC-based quick portal authentication for portal users on the interface.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Specify a MAC binding server on the interface.

   `portal apply mac-trigger-server` *`server-name`*

   By default, no MAC binding server is specified on an interface.

Configure cloud MAC-trigger authentication

About this task

This feature enables the cloud portal server to act as a MAC binding server to perform cloud MAC-trigger authentication on portal users.

Procedure

1. Enter system view.

   `system-view`

2. Create a MAC binding server and enter its view.

   `portal mac-trigger-server` *`server-name`*

3. Enable cloud MAC-trigger authentication.

   `cloud-binding enable`

   By default, cloud MAC-trigger authentication is disabled.

4. Specify the URL of the cloud portal authentication server.

   `cloud-server url` *`url-string`*

   By default, the cloud portal authentication server URL is not specified. The device uses the URL of the portal Web server as the URL of the cloud portal authentication server.

Logging out online portal users

About this task

This feature deletes users that have passed portal authentication and terminates ongoing portal authentications.

Restrictions and guidelines

If any device in a VSRP group executes the **`portal delete-user`** command to log out a user, all devices in this group log out the user.

When the number of online users exceeds 2000, executing the **portal delete-user** command takes a few minutes.

To ensure successful logout of online users, do not perform the following operations during the command execution:

- Master/backup device switchover.
- Breaking the data channel in the VSRP group.
- Active/standby MPU switchover.
- Disabling portal authentication on the interface.

Procedure

1. Enter system view.

   system-view

2. Log out online portal users.

   IPv4:

   portal delete-user { *ipv4-address* | **all** | **auth-type** { **cloud** | **email** | **facebook** | **local** | **normal** | **qq** | **wechat** } | **interface** *interface-type interface-number* | **mac** *mac-address* | **username** *username* }

   IPv6:

   portal delete-user { **all** | **auth-type** { **cloud** | **email** | **facebook** | **local** | **normal** | **qq** | **wechat** } | **interface** *interface-type interface-number* | **ipv6** *ipv6-address* | **mac** *mac-address* | **username** *username* }

Setting the user traffic backup threshold

About this task

The device backs up traffic for a user when the user's traffic reaches the user traffic backup threshold. A smaller threshold provides more accurate backup for user traffic. However, when a large number of users exist, a small threshold results in frequent user traffic backups, affecting the user online, offline, and accounting processes. Set a proper threshold to balance between service performance and traffic backup accuracy.

Procedure

1. Enter system view.

   system-view

2. Set the user traffic backup threshold.

   portal traffic-backup threshold *value*

   By default, the user traffic backup threshold is 10 MB.

Disabling the Rule ARP or ND entry feature for portal clients

About this task

When the Rule ARP or ND entry feature is enabled for portal clients, ARP or ND entries for portal clients are Rule entries after the clients come online. The Rule entries will not age out and will be deleted immediately after the portal clients go offline. If a portal client goes offline and then tries to get online before the ARP or ND entry is relearned for the client, the client will fail the authentication. To avoid such authentication failure, disable this feature. Then, ARP or ND entries for portal clients are dynamic entries after the clients come online and are deleted only when they age out.

Restrictions and guidelines

Enabling or disabling of this feature does not affect existing Rule/dynamic ARP or ND entries.

Procedure

1. Enter system view.
   system-view
2. Disable the Rule ARP or ND entry feature for portal clients.
   undo portal refresh { **arp** | **nd** } **enable**
   By default, the Rule ARP or ND entry feature is disabled.

Disabling traffic accounting for portal users

About this task

The accounting server might perform time-based or traffic-based accounting, or it might not perform accounting.

If the accounting server does not perform traffic-based accounting, disable traffic accounting for portal users on the device. The device will provide quick accounting for portal users, and the traffic statistics will be imprecise.

If the accounting server performs traffic-based accounting, enable traffic accounting for portal users. The device will provide precise traffic statistics for portal users.

Procedure

1. Enter system view.
   system-view
2. Disable traffic accounting for portal users.
   portal traffic-accounting disable
   By default, traffic accounting is enabled for portal users.

Configuring Web redirect

About Web redirect

Web redirect is a simplified portal feature. With Web redirect, a user does not perform portal authentication but is directly redirected to the specified URL on the first Web access attempt in a browser. After the specified redirect interval, the user is redirected from the visiting website to the specified URL again.

Web redirect can provide ISPs with extended services. For example, the ISPs can place advertisements and publish information on the redirected webpage.

Configuring Web redirect on an interface

Restrictions and guidelines

The Web redirect feature takes effect only on HTTP packets that use the default port number 80.

To use the device URL as the Web redirect URL or allow users to successfully access the device URL, you must enable the HTTP service. To enable the HTTP service, use the **ip http enable** command.

When Web redirect and portal authentication are all enabled on an interface, the device redirects users on the interface as follows:

- The device redirects the first HTTP request of a user to the specified URL. Then, the device redirects the next HTTP request of the user to the portal authentication page. After the user logs out, the user is redirected to the specified URL again at the first Web access.
- After the specified redirect interval, a user is redirected to the specified URL regardless of whether the user is online or not. This process does not cause online users to be offline.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Configure Web redirect.

   web-redirect [**ipv6**] **url** *url-string* [**interval** *interval*]

   By default, Web redirect is disabled.

Configuring destination-based portal redirection rules

About this task

The device uses destination-based portal redirection rules to perform URL redirection. If the Web request of a portal user matches the specified destination in a redirection rule, the device redirects the user to the URL specified in the redirection rule.

Restrictions and guidelines

If the Web request of a portal user matches a destination-based portal redirection rule and a URL redirection match rule (configured by using the **if-match** command), the redirection rule takes effect.

If you specify a host name or IP address in a destination-based portal redirection rule, do not specify a URL that includes the host name or IP address as the redirection URL in another rule. A violation will cause redirect loops.

The system supports a maximum of 10 destination-based portal redirection rules. For the same host or IP address, only one destination-based portal redirection rule is supported.

Procedure

1. Enter system view.

   system-view

2. Configure a destination-based portal redirection rule.

   portal redirect-rule destination { **host** { *host-name* | *ip-address* } | **ipv6** *ipv6-address* } [**redirect-url** *url*]

   By default, no destination-based portal redirection rules are configured.

Configuring portal safe-redirect

About this task

Portal safe-redirect filters HTTP requests by HTTP request method, browser type (in HTTP User Agent), and destination URL, and redirects only the permitted HTTP requests. This reduces the risk that the portal Web server cannot respond to HTTP requests because of overload.

Table 2 Browser types supported by portal safe-redirect

Browser type	Description
Safari	Apple browser
Chrome	Google browser
Firefox	Firefox browser
UC	UC browser
QQBrowser	QQ browser
LBBROWSER	Cheetah browser
TaoBrowser	Taobao browser
Maxthon	Maxthon browser
BIDUBrowser	Baidu browser
MSIE 10.0	Microsoft IE 10.0 browser
MSIE 9.0	Microsoft IE 9.0 browser
MSIE 8.0	Microsoft IE 8.0 browser
MSIE 7.0	Microsoft IE 7.0 browser
MSIE 6.0	Microsoft IE 6.0 browser
MetaSr	Sogou browser

Procedure

1. Enter system view.

   `system-view`

2. Enable portal safe-redirect.

   `portal safe-redirect enable`

   By default, the portal safe-redirect feature is disabled.

3. (Optional.) Specify HTTP request methods permitted by portal safe-redirect.

   `portal safe-redirect method` { **`get`** | **`post`** }

   By default, the device can redirect only HTTP requests with GET method after portal safe-redirect is enabled.

4. (Optional.) Specify a browser type permitted by portal safe-redirect.

   `portal safe-redirect user-agent` *`user-agent-string`*

   By default, no browser types are specified. The device can redirect HTTP requests sent by all supported browsers (see Table 2) after portal safe-redirect is enabled.

5. (Optional.) Configure a URL forbidden by portal safe-redirect.

   `portal safe-redirect user-agent` *`user-agent-string`*

   By default, no forbidden URLs are configured. The device can redirect HTTP requests with any URLs.

6. (Optional.) Configure a filename extension forbidden by portal safe-redirect.

   `portal safe-redirect forbidden-url` *`user-url-string`*

   By default, no forbidden filename extensions are configured. The device redirects HTTP requests regardless of the file extension in the URL.

7. (Optional.) Configure a URL keyword forbidden by portal safe-redirect.

portal safe-redirect forbidden-keyword *keyword*

By default, no forbidden URL keywords are configured. The device redirects HTTP requests regardless of the keywords in the URL.

8. (Optional.) Configure the default action for portal safe-redirect.

   portal safe-redirect default-action { **forbidden** | **permit** }

   By default, no default action is configured for portal safe-redirect.

9. Configure a URL permitted by portal safe-redirect.

   portal safe-redirect permit-url *user-url-string*

   By default, the device can redirect Web requests that contain any URLs.

Setting the maximum number of portal redirection sessions for a single user

About this task

If a user client is attacked by malicious software or viruses, it might initiate a large number of portal redirect sessions. You can perform this task to limit the number of portal redirect sessions that can be established for that user.

The maximum number applies to the HTTP redirect sessions and HTTPS redirect sessions separately. For example, assume you set the maximum number to 50. Then, a portal user can establish a maximum of 50 HTTP redirect sessions and a maximum of 50 HTTPS redirect sessions.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of portal redirection sessions for a single user.

   portal redirect max-session per-user *number*

   By default, no limit is set on the number of portal redirect sessions for a single user.

Excluding an attribute from portal protocol packets

About this task

Support of the portal authentication server for portal protocol attributes varies by the server type. If the device sends the portal authentication server a packet that contains an attribute unsupported by the server, the device and the server cannot communicate.

To address this issue, you can configure portal protocol packets to not carry the attributes unsupported by the portal authentication server.

Excluding an attribute from portal protocol packets for a portal authentication server

1. Enter system view.

   system-view

2. Enter portal authentication server view.

   portal server *server-name*

3. Exclude an attribute from portal protocol packets.

   exclude-attribute *number* { **ack-auth** | **ntf-logout** | **ack-logout** }

   By default, no attributes are excluded from portal protocol packets.

Excluding an attribute from portal protocol packets for a MAC binding server

1. Enter system view.

   system-view
2. Enter MAC binding server view.

   portal mac-trigger-server *server-name*
3. Exclude an attribute from portal protocol packets.

   exclude-attribute *attribute-number*

   By default, no attributes are excluded from portal protocol packets.

Configuring support of portal authentication for third-party authentication

About third-party authentication

The device supports using a third-party authentication server, such as QQ, email, WeChat, or Facebook authentication server as the portal authentication server to complete portal authentication. No portal authentication servers are required to be deployed, and no local portal users are required to be created on the device. This reduces the management and maintenance cost.

You need to add a third-party authentication button on the portal authentication page. A user clicks the button and is redirected to the third-party authentication page. The user then uses the third-party authentication account to perform portal authentication.

Restrictions and guidelines for third-party authentication

Only direct portal authentication that uses a local portal Web service supports third-party authentication.

Editing buttons and pages for third-party authentication

Restrictions and guidelines

No authentication button or authentication page is required for WeChat authentication.

Editing a third-party authentication button

To provide QQ, email, or Facebook authentication for portal users, you must add a QQ, email, or Facebook authentication button to the portal logon page.

When you edit the QQ authentication button, you must call the **pt_getQQSubmitUrl()** function to get the URL of the QQ authentication page. The following example shows part of the script of the QQ authentication button.

```
    <html>
    <head>
    <title>Logon</title>
    <script type="text/javascript" language="javascript" src="pt_private.js"></script>
<script type="text/javascript">
      function setQQUrl(){
          document.getElementById("qqurl").href = pt_getQQSubmitUrl();
            }
</script>
```

```
    </head>
    <body>
    ... ...
<a href="javascript:void(null)" id="qqurl" onclick="setQQUrl()">QQ</a>
    ... ...
    </body>
</html>
```

No special requirements exist in the process of editing an email or Facebook authentication button.

Editing a third-party authentication page

You need to edit the email authentication page and the Facebook authentication page. The QQ authentication page is provided by Tencent.

When you edit the email authentication page, follow the rules in "Customizing authentication pages" and the following rules:

- Set the action attribute of the beginning form tag to **maillogin.html**. Otherwise, the device cannot send the user information
- Save the login page as **emailLogon.htm**.

The following example shows part of the script of the **emailLogon.htm** page.

```
<form action= maillogin.html method = post >
<p>User name:<input type="text" name = "PtUser" style="width:160px;height:22px"
maxlength=64>
<p>Password :<input type="password" name = "PtPwd" style="width:160px;height:22px"
maxlength=32>
<p><input type=SUBMIT value="Logon" name = "PtButton" style="width:60px;"
onclick="form.action=form.action+location.search;">
</form>
```

When you edit the Facebook authentication page, follow the rules in "Customizing authentication pages."

Configuring QQ authentication

About this task

After a portal user passes QQ authentication, the QQ authentication server sends the authorization code of the user to the portal Web server. After the portal Web server receives the authorization code, it sends the authorization code of the user, the app ID, and the app key to the QQ authentication server for verification. If the information is verified as correct, the device determines that the user passes QQ authentication.

Prerequisites

To provide QQ authentication for portal users, you must go to Tencent Open Platform (**http://connect.qq.com/intro/login**) to finish the following tasks:

1. Register as a developer by using a valid QQ account.
2. Apply the access to the platform for your website. The website is the webpage to which users are redirected after passing QQ authentication.

You will obtain the app ID and app key from the Tencent Open Platform after your application succeeds.

Procedure

1. Enter system view.

   `system-view`

2. Create a QQ authentication server and enter its view.

   portal extend-auth-server qq

3. (Optional.) Specify the URL of the QQ authentication server.

   auth-url *url-string*

   By default, the URL of a QQ authentication server is **https://graph.qq.com**.

4. (Optional.) Specify the URL to which portal users are redirected after they pass QQ authentication.

   redirect-url *url-string*

   By default, portal users are redirected to URL **http://oauthindev.nsfocus.com.cn/portal/qqlogin.html** after they pass QQ authentication.

5. (Optional.) Specify the app ID for QQ authentication.

   app-id *app-id*

   By default, an app ID for QQ authentication exists.

6. (Optional.) Specify the app key for QQ authentication.

   app-key *app-key*

   By default, an app key for QQ authentication exists.

Configuring email authentication

About this task

If a portal user chooses email authentication, the user can access the network after passing email authentication.

Procedure

1. Enter system view.

   system-view

2. Create an email authentication server and enter its view.

   portal extend-auth-server mail

3. Specify protocols for email authentication.

   mail-protocol { **imap** | **pop3** } *

   By default, no protocols are specified for email authentication.

4. Specify an email domain name for email authentication.

   mail-domain-name *string*

   By default, no email domain names are specified for email authentication.

Configuring WeChat authentication

About this task

During WeChat authentication, the device first sends the credentials (app ID, app key, and shop ID) for WeChat authentication to the WeChat Official Account Platform for verification. After the credentials are verified, the device continues the portal authentication and allows the user to use the WiFi network after the authentication.

The subscribe-required feature requires users to follow the WeChat official account during WeChat authentication. If the users do not follow the WeChat official account, they fail WeChat authentication.

When subscribe-required feature is configured, the device sends the app ID and app secret to the WeChat Official Account Admin Platform to obtain the access token. Upon receiving authentication requests from portal users, the device sends the access token and the open ID in the authentication requests to the WeChat server to obtain user information. Based on the returned user information, the device determines whether the portal users have followed the WeChat official account.

Prerequisites

Before you configure WeChat authentication, you must go to the WeChat Official Account Admin Platform (**https://mp.weixin.qq.com**) to finish the following tasks:

1. Apply a WeChat official account.
2. Use the account to log in to the platform and enable the WeChat WiFi hotspot feature.
3. Click the device management tab, add the device: select the shop where the device is deployed, select the **portal** device type, and enter the device settings.

   After the previous configurations, you will obtain the credentials (app ID, app key, and shop ID) for WeChat authentication.

To obtain the app secret for WeChat authentication, perform the following tasks:

1. Use the WeChat official account to log in to the WeChat Official Account Admin Platform.
2. From the navigation tree, select **Developer Centers**.

   In the **Configuration Items** area, you can see the app secret for the WeChat Official account.

Procedure

1. Enter system view.

   `system-view`
2. Create a WeChat authentication server and enter its view.

   `portal extend-auth-server wechat`
3. (Optional.) Specify the app ID for WeChat authentication.

   `app-id` *`app-id`*

   By default, no app ID is specified for WeChat authentication.
4. (Optional.) Specify the app key for WeChat authentication.

   `app-key` *`app-key`*

   By default, no app key is specified for WeChat authentication.
5. (Optional.) Specify the shop ID for WeChat authentication.

   `shop-id` *`shop-id`*

   By default, no app key is specified for WeChat authentication.
6. (Optional.) Configure the subscribe-required feature.
   a. Enable the subscribe-required feature:

      `subscribe-required enable`

      By default, the subscribe-required feature is disabled.

      This feature must be used with the portal temporary pass feature. As a best practice, set the temporary pass period to 600 seconds.

   b. Specify the app secret for WeChat authentication.

      `app-secret` { **`cipher`** | **`simple`** } *`string`*

      By default, no app secret is specified for WeChat authentication.

Configuring Facebook authentication

Prerequisites

To use Facebook authentication for portal users, you must register as a developer on the Facebook website to obtain an app ID and app key.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Create a Facebook authentication server and enter its view.

   ```
   portal extend-auth-server facebook
   ```

3. (Optional.) Specify the URL of the Facebook authentication server.

   ```
   auth-url url-string
   ```

   By default, the URL of Facebook authentication server is **https://graph.facebook.com**.

4. (Optional.) Specify the URL to which portal users are redirected after they pass Facebook authentication.

   ```
   redirect-url url-string
   ```

   By default, portal users are redirected to URL **http://oauthindev.nsfocus.com.cn/portal/fblogin.html** after they pass Facebook authentication.

5. (Optional.) Specify the app ID for Facebook authentication.

   ```
   app-id app-id
   ```

   By default, no app ID is specified for Facebook authentication.

6. (Optional.) Specify the app key for Facebook authentication.

   ```
   app-key app-key
   ```

   By default, no app key is specified for Facebook authentication.

Specifying an authentication domain for third-party authentication

About this task

Specify an authentication domain for third-party authentication on an interface to apply the authentication, authorization, and accounting methods in the domain to portal users.

Restrictions and guidelines

Make sure the authentication, authorization, and accounting methods in the specified authentication domain are **none**.

Specifying an authentication domain for third-party authentication on an interface

1. Enter system view.

   ```
   system-view
   ```

2. Enter Layer 3 interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify an authentication domain for third-party authentication on the interface.

   ```
   portal extend-auth domain domain-name
   ```

   By default, no authentication domain is specified for third-party authentication on an interface.

Configuring portal temporary pass

About this task

Typically, a portal user cannot access the Internet before passing portal authentication. This feature allows a user to access the Internet temporarily if the user uses a WeChat account to perform portal authentication. During the temporary pass period, the user can provide WeChat authentication information to the WeChat server for the server to interact with the access device to finish portal authentication.

Restrictions and guidelines

This feature is available only for direct portal authentication mode.

If both portal safe-redirect and portal temporary pass match rules are configured, portal temporary pass match rules take precedence.

Configuring portal temporary pass on an interface

1. Enter system view.
   `system-view`
2. Enter Layer 3 interface view.
   `interface` *`interface-type`* *`interface-number`*
3. Enable portal temporary pass and set the temporary pass period on the interface.
   `portal temp-pass` [**`period`** *`period-value`*] **`enable`**
   By default, portal temporary pass is disabled on an interface.
4. Configure a match rule for portal temporary pass:
   a. Return to system view.
      `quit`
   b. Enter portal Web server view.
      `portal web-server` *`server-name`*
   c. Configure a match rule for portal temporary pass.
      `if-match` { **`original-url`** *`url-string`* | **`user-agent`** *`user-agent`* } * **`temp-pass`** [**`redirect-url`** *`url-string`* | **`original`**]
      By default, no match rules for portal temporary pass are configured.

Setting the user synchronization interval for portal authentication using OAuth

About this task

If portal authentication uses OAuth, the device periodically reports user information to the portal authentication server for user synchronization on the server. To disable user synchronization from the device to the portal authentication server, set the user synchronization interval to 0 seconds on the device.

Procedure

1. Enter system view.
   `system-view`
2. Set the user synchronization interval for portal authentication using OAuth.
   `portal oauth user-sync interval` *`interval`*
   By default, the user synchronization interval is 60 seconds.

Configuring user synchronization for portal authentication using the WiFiDog protocol

About this task

Use this feature when users perform portal authentication using the WiFiDog protocol. This feature enables the device to periodically synchronize user information with the portal server to ensure user information consistency between the device and the portal server.

Restrictions and guidelines

For this feature to take effect, make sure the type of the portal Web server is WiFiDog before you perform this task. To specify the type of the portal Web server, use the **server-type** command.

Procedure

1. Enter system view.

   system-view

2. Enable user information synchronization and set the synchronization interval for portal authentication using WiFiDog.

   portal wifidog user-sync interval *interval*

   By default, user information synchronization is disabled for portal authentication using WiFiDog.

Configuring portal authentication information report interval

About this task

After you configure this feature, the device reports portal authentication failure and error information to the cloud server. The first report is sent to the cloud server 30 seconds after the device is connected to the server. The subsequent reports are sent at regularly intervals as configured by this feature.

If you modify the report interval, the modified interval takes effect on the next report.

Procedure

1. Enter system view.

   system-view

2. Configure the time interval at which portal authentication information is reported to the cloud server.

   portal cloud report interval *minutes*

   By default, the portal authentication information is reported to the cloud server at intervals of 5 minutes.

Enabling portal logging

About this task

To help with security audits, you can enable portal logging to record portal authentication information.

For portal log messages to be sent correctly, you must also configure the information center on the device. For more information about information center configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable logging for portal user logins and logouts.

   portal user log enable

   By default, portal user login and logout logging is disabled.

3. Enable logging for portal protocol packets.

   portal packet log enable

   By default, portal protocol packet logging is disabled.

4. Enable logging for portal redirect.

   portal redirect log enable

   By default, portal redirect logging is disabled.

Configuring the portal authentication monitoring feature

About this task

The portal authentication monitoring feature records portal user offlines, authentication failures, and authentication errors. These records help the administrator quickly identify causes of authentication faults.

Procedure

1. Enter system view.

   system-view

2. Enable portal user offline recording.

   portal logout-record enable

   By default, portal user offline recording is disabled.

3. Set the maximum number of portal user offline records.

   portal logout-record max *number*

   The default setting varies by device model. For more information, see the command reference.

4. Export portal user offline records to a path.

   portal logout-record export url *url-string* [**start-time** *start-date start-time* **end-time** *end-date end-time*]

5. Enable portal authentication failure recording.

   portal auth-fail-record enable

   By default, portal authentication failure recording is enabled.

6. Set the maximum number of portal authentication failure records.

   portal auth-fail-record max *number*

   The default setting varies by device model. For more information, see the command reference.

7. Export portal authentication failure records to a path.

   portal auth-fail-record export url *url-string* [**start-time** *start-date start-time* **end-time** *end-date end-time*]

8. Enable portal authentication error recording.

   portal auth-error-record enable

   By default, portal authentication error recording is enabled.

9. Set the maximum number of portal authentication error records.

   portal auth-error-record max *number*

   The default setting varies by device model. For more information, see the command reference.

10. Export portal authentication error records to a path.

    portal auth-error-record export url *url-string* [**start-time** *start-date start-time* **end-time** *end-date end-time*]

Configuring portal advertisement push

About this task

This feature enables the device to push advertisements to portal users after they pass portal authentication.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable portal advertisement push.

   portal ad-push enable

   By default, portal advertisement push is disabled.

4. Configure the advertisement URL or advertisement group to be pushed to portal users.

   portal [**ipv6**] **ad-push** { **url** *url-string* [**interval** *interval* | **time-range** *time-range-name* | **traffic-threshold** *traffic-threshold*] | **url-group** *group-name* }

   By default, no advertisement URL or advertisement group is configured.

5. (Optional.) Return to system view.

   quit

6. (Optional.) Configure a portal advertisement whitelist.

   portal ad-push whitelist { **ip** *ipv4-address* | **ipv6** *ipv6-address* | **mac-address** *mac-address* }

   By default, no portal advertisement whitelist is configured.

7. (Optional.) Create an advertisement group and enter its view.

   portal ad-url-group *group-name* [**method** { **interval** | **time-range** | **traffic** }]

8. (Optional.) Configure an advertisement URL in the advertisement group.

   ad-url *url-string* [**time-range** *time-range-name*]

   By default, no advertisement URLs are configured in an advertisement group.

9. (Optional.) Set the time interval or traffic threshold for portal advertisement push in the advertisement group.

   ad-url-group { **interval** *interval* | **traffic-threshold** *traffic-threshold* }

By default, the interval is 360 minutes and the traffic threshold is 100 MB for portal advertisement push in an advertisement group.

Display and maintenance commands for portal

Execute **display** commands in any view and the **reset** command in user view.

Task	Command
Display portal configuration and portal running state.	**display portal interface** *interface-type interface-number*
Display statistics about portal advertisement push.	**display portal ad-push statistics** { **ad-url-group** | **url** }
Display portal authentication error records.	**display portal auth-error-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time* **end-time** *end-date end-time* }
Display portal authentication failure records.	**display portal auth-fail-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time* **end-time** *end-date end-time* | **username** *username* }
Display packet statistics for portal captive-bypass.	**display portal captive-bypass statistics** [**slot** *slot-number*]
Display IP addresses corresponding to host names in destination-based portal-free rules.	**display portal dns free-rule-host** [*host-name*]
Display IP addresses resolved by host names in destination-based portal redirection rules.	**display portal dns redirect-rule-host** [*host-name*]
Display information about third-party authentication servers.	**display portal extend-auth-server** { **all** | **facebook** | **mail** | **qq** | **wechat** }
Display information about local MAC-account binding entries.	**display portal local-binding mac-address** { **all** | *mac-address* }
Display portal user offline records.	**display portal logout-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time* **end-time** *end-date end-time* | **username** *username* }
Display information about MAC-trigger authentication users (portal users that perform MAC-trigger authentication).	**display portal mac-trigger user** { **all** | **ip** *ipv4-address* | **mac** *mac-address* }
Display information about MAC binding servers.	**display portal mac-trigger-server** { **all** | **name** *server-name* }
Display packet statistics for portal authentication servers.	**display portal packet statistics** [**extend-auth-server** { **cloud** | **facebook** | **mail** | **qq** | **wechat** } | **mac-trigger-server** *server-name* | **server** *server-name*]
Display statistics for portal permit rules.	**display portal permit-rule statistics**

Task	Command
Display portal redirect session information	**display portal redirect session** [**ip** *ipv4-address* | **ipv6** *ipv6-address*] [**slot** *slot-number*]
Display history records about portal redirect sessions.	**display portal redirect session-record** [**start-time** *start-date start-time*] [**end-time** *end-date end-time*] [**slot** *slot-number*]
Display summary statistics about portal redirect sessions.	**display portal redirect session-statistics** [**slot** *slot-number*]
Display portal redirect packet statistics.	**display portal redirect statistics** [**slot** *slot-number*]
Display portal rules.	**display portal rule** { **all** | **dynamic** | **static** } **interface** *interface-type interface-number* [**slot** *slot-number*]
Display packet statistics for portal safe-redirect	**display portal safe-redirect statistics** [**slot** *slot-number*]
Display portal authentication server information.	**display portal server** [*server-name*]
Display portal user information.	**display portal user** { **all** | **auth-type** { **cloud** | **email** | **facebook** | **local** | **mac-trigger** | **normal** | **qq** | **wechat** } | **interface** *interface-type interface-number* | **ip** *ip-address* | **ipv6** *ipv6-address* | **mac** *mac-address* | **pre-auth** [**interface** *interface-type interface-number* | **ip** *ip-address* | **ipv6** *ipv6-address*] | **username** *username* } [**brief** | **verbose**]
Display the number of portal users.	**display portal user count**
Display DHCP lease information of IPv4 portal users.	**display portal user dhcp-lease** [**ipv4** *ipv4-address*]
Display DHCPv6 lease information of IPv6 portal users.	**display portal user dhcpv6-lease** [**ipv6** *ipv6-address*]
Display information about portal users blocked for authentication failure.	**display portal user-block** [**ip** *ipv4-address* | **ipv6** *ipv6-address* | **mac** *mac-address* | **username** *username*]
Display portal Web server information.	**display portal web-server** [*server-name*]
Display Web redirect rule information.	**display web-redirect rule interface** *interface-type interface-number* [**slot** *slot-number*]
Clear statistics about portal advertisement push.	**reset portal ad-push statistics** { **ad-url-group** | **url** }
Clear portal authentication error records.	**reset portal auth-error-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time*

Task	Command
	end-time *end-date end-time* }
Clear portal authentication failure records.	**reset portal auth-fail-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time* **end-time** *end-date end-time* | **username** *username* }
Clear packet statistics for portal captive-bypass.	**reset portal captive-bypass statistics** [**slot** *slot-number*]
Clear local MAC-account binding entries.	**reset portal local-binding mac-address** { *mac-address* | **all** }
Clear portal user offline records.	**reset portal logout-record** { **all** | **ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **start-time** *start-date start-time* **end-time** *end-date end-time* | **username** *username* }
Clear packet statistics for portal authentication servers.	**reset portal packet statistics** [**extend-auth-server** { **cloud** | **facebook** | **mail** | **qq** | **wechat** } | **mac-trigger-server** *server-name* | **server** *server-name*]
Clear history records about portal redirect sessions.	**reset portal redirect session-record** [**slot** *slot-number*]
Clear summary statistics for portal redirect sessions.	**reset portal redirect session-statistics** [**slot** *slot-number*]
Clear portal redirect packet statistics.	**reset portal redirect statistics** [**slot** *slot-number*]
Clear packet statistics for portal safe-redirect.	**reset portal safe-redirect statistics** [**slot** *slot-number*]

Portal configuration examples

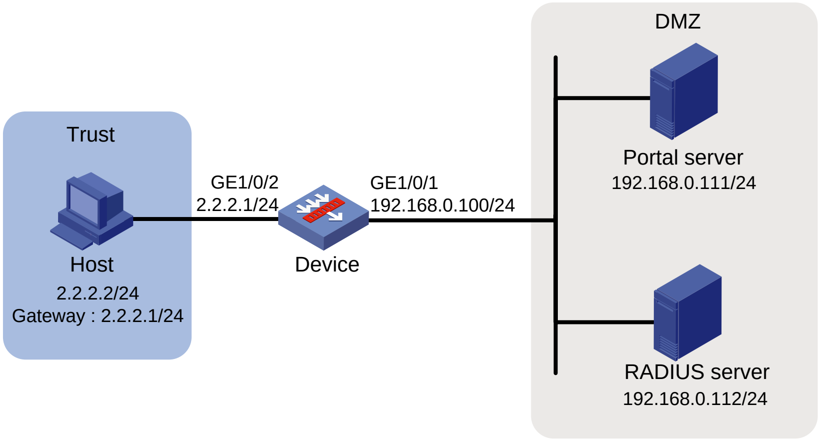
Example: Configuring direct portal authentication

Network configuration

As shown in Figure 6, the host is directly connected to the device (the access device). The host is assigned a public IP address either manually or through DHCP. An IMC server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server. In this example, the IMC server runs IMC 7.1 (E0303) and IMC UAM 7.1 (E0304).

Configure direct portal authentication, so the host can access only the portal server before passing the authentication and access other network resources after passing the authentication.

Figure 6 Network diagram



Configuring the RADIUS server

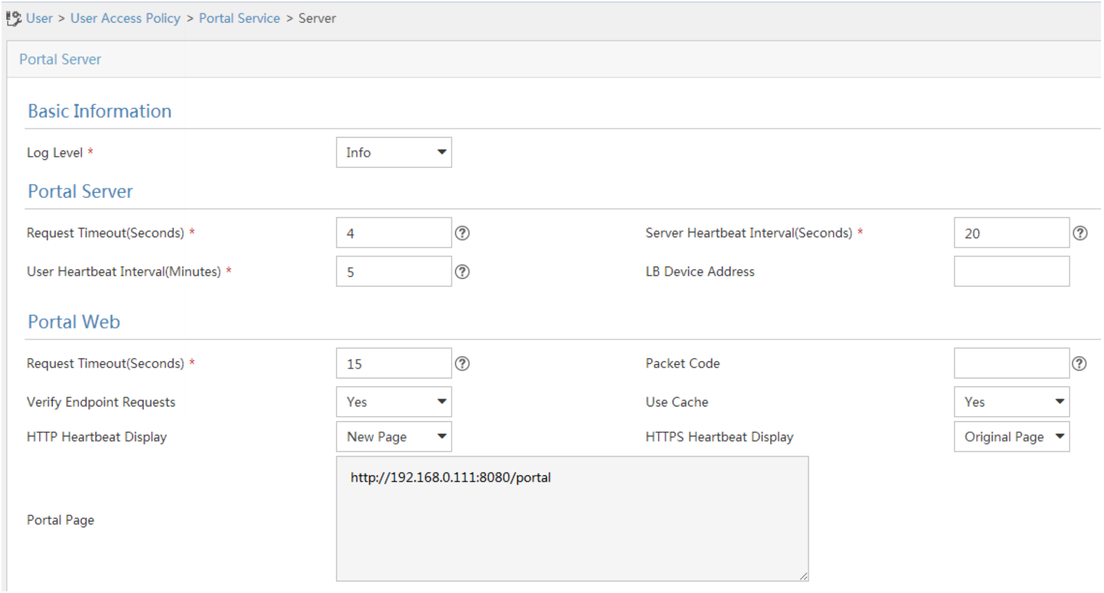
# Configure the RADIUS server correctly to provide authentication and accounting functions. (Details not shown.)

Configuring the portal server

1. Configure the portal authentication server:
   a. Log in to IMC and click the **User** tab.
   b. Select **User Access Policy** > **Portal Service** > **Server** from the navigation tree to open the portal server configuration page, as shown in Figure 7.
   c. Configure the portal server parameters as needed.

      This example uses the default settings.
   d. Click **OK**.

Figure 7 Portal server configuration



User > User Access Policy > Portal Service > Server

Portal Server

Basic Information

Log Level * Info

Portal Server

Request Timeout(Seconds) * 4

Server Heartbeat Interval(Seconds) * 20

User Heartbeat Interval(Minutes) * 5

LB Device Address

Portal Web

Request Timeout(Seconds) * 15

Packet Code

Verify Endpoint Requests Yes

Use Cache Yes

HTTP Heartbeat Display New Page

HTTPS Heartbeat Display Original Page

Portal Page http://192.168.0.111:8080/portal

2. Configure the IP address group:
   a. Select **User Access Policy** > **Portal Service** > **IP Group** from the navigation tree to open the portal IP address group configuration page.

b. Click **Add** to open the page as shown in Figure 8.

c. Enter the IP group name.

d. Enter the start IP address and end IP address of the IP group.

   Make sure the host IP address is in the IP group.

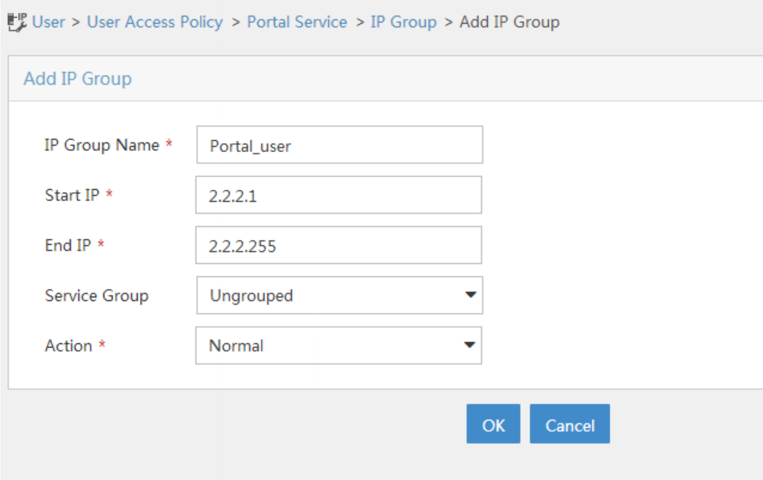
e. Select a service group.

   This example uses the default group **Ungrouped**.

f. Select **Normal** from the **Action** list.

g. Click **OK**.

Figure 8 Adding an IP address group



User > User Access Policy > Portal Service > IP Group > Add IP Group

Add IP Group

IP Group Name *	Portal_user
Start IP *	2.2.2.1
End IP *	2.2.2.255
Service Group	Ungrouped
Action *	Normal

3. Add a portal device:

   a. Select **User Access Policy** > **Portal Service** > **Device** from the navigation tree to open the portal device configuration page.

   b. Click **Add** to open the page as shown in Figure 9.

   c. Enter device name **NAS**.

   d. Enter the IP address of the device's interface connected to the host.

   e. Select whether to support server heartbeat and user heartbeat functions.

      In this example, select **No** for both **Support Server Heartbeat** and **Support User Heartbeat**.

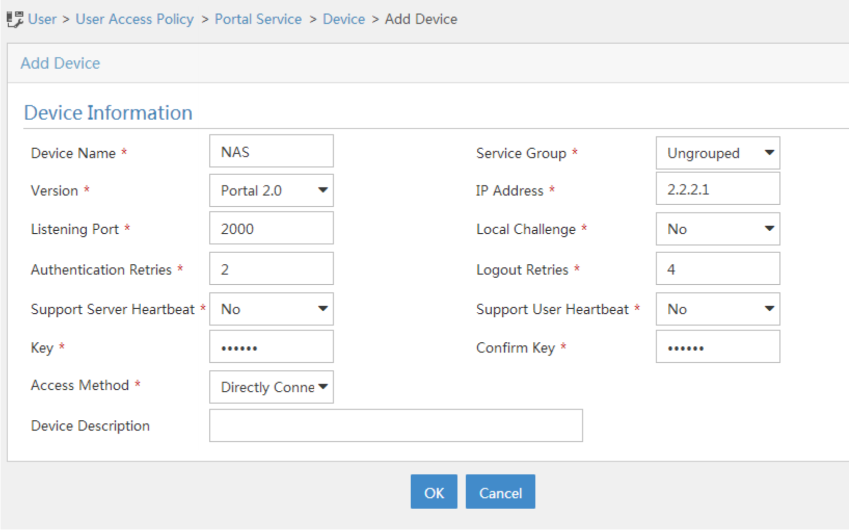
   f. Enter the key, which must be the same as that configured on the device.

   g. Select **Directly Connected** from the **Access Method** list.

   h. Use the default settings for other parameters.

   i. Click **OK**.

Figure 9 Adding a portal device



User > User Access Policy > Portal Service > Device > Add Device

Add Device

Device Information

Device Name *	NAS	Service Group *	Ungrouped
Version *	Portal 2.0	IP Address *	2.2.2.1
Listening Port *	2000	Local Challenge *	No
Authentication Retries *	2	Logout Retries *	4
Support Server Heartbeat *	No	Support User Heartbeat *	No
Key *	••••••	Confirm Key *	••••••
Access Method *	Directly Conne		
Device Description			

4. Associate the portal device with the IP address group:

   a. As shown in Figure 10, click the **Port Group Information Management** icon for device **NAS** to open the port group configuration page.

   b. Click **Add** to open the page as shown in Figure 11.

   c. Enter the port group name.

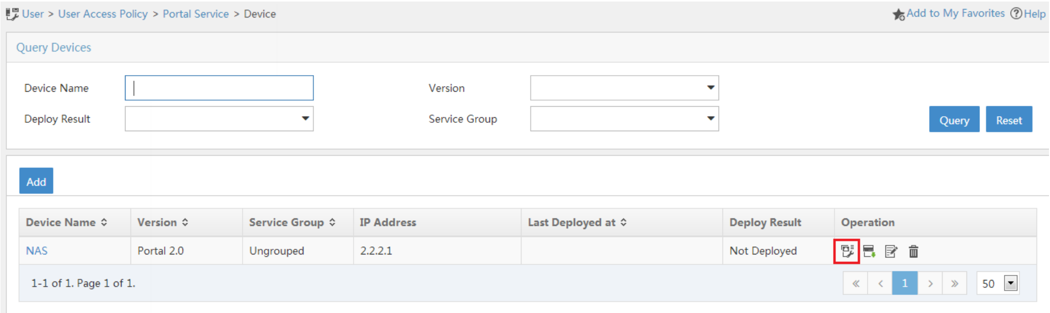
   d. Select the configured IP address group.

      The IP address used by the user to access the network must be within this IP address group.

   e. Use the default settings for other parameters.

   f. Click **OK**.

Figure 10 Device list

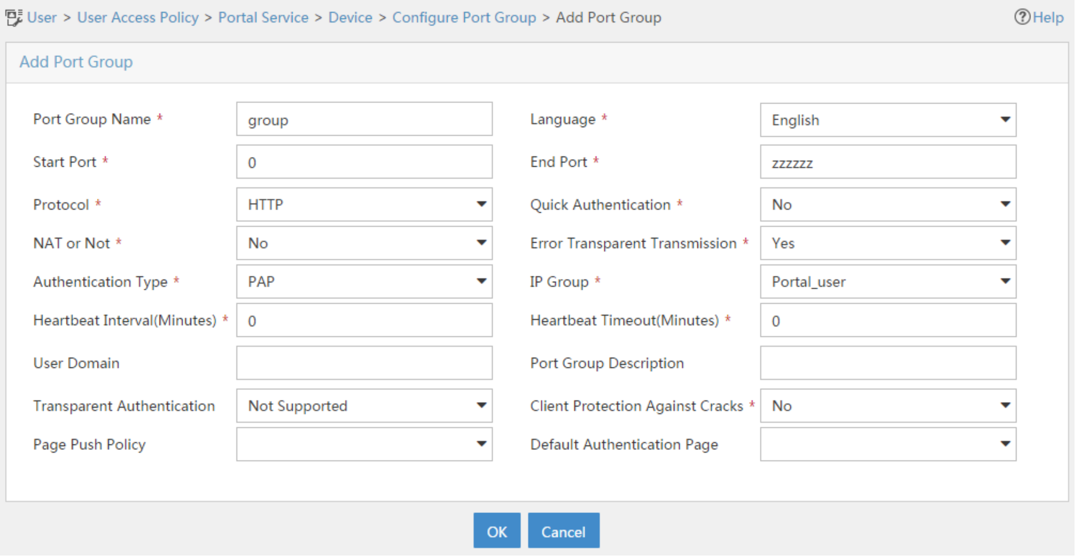


User > User Access Policy > Portal Service > Device

Device Name	Version	Service Group	IP Address	Last Deployed at	Deploy Result	Operation
NAS	Portal 2.0	Ungrouped	2.2.2.1		Not Deployed	

1-1 of 1. Page 1 of 1.

Figure 11 Adding a port group


User > User Access Policy > Portal Service > Device > Configure Port Group > Add Port Group
Help
Add Port Group
OK
Cancel

Port Group Name *	group	Language *	English
Start Port *	0	End Port *	zzzzzz
Protocol *	HTTP	Quick Authentication *	No
NAT or Not *	No	Error Transparent Transmission *	Yes
Authentication Type *	PAP	IP Group *	Portal_user
Heartbeat Interval(Minutes) *	0	Heartbeat Timeout(Minutes) *	0
User Domain		Port Group Description	
Transparent Authentication	Not Supported	Client Protection Against Cracks *	No
Page Push Policy		Default Authentication Page	

Configuring the device

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
[Device-security-zone-DMZ] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-1-trust-dmz] source-zone trust
[Device-security-policy-ip-1-trust-dmz] destination-zone dmz
[Device-security-policy-ip-1-trust-dmz] source-ip-host 2.2.2.2
[Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-1-trust-dmz] action pass
[Device-security-policy-ip-1-trust-dmz] quit
```

   # Configure a rule named **portallocalout** to allow the device to send packets to the RADIUS server and portal server.

```
[Device-security-policy-ip] rule name portallocalout
[Device-security-policy-ip-2-portallocalout] source-zone local
[Device-security-policy-ip-2-portallocalout] destination-zone dmz
```

```
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[Device-security-policy-ip-2-portallocalout] action pass
[Device-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow the device to receive and process packets from the RADIUS server and portal server.

```
[Device-security-policy-ip] rule name portallocalin
[Device-security-policy-ip-3-portallocalin] source-zone dmz
[Device-security-policy-ip-3-portallocalin] destination-zone local
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin] action pass
[Device-security-policy-ip-3-portallocalin] quit
[Device-security-policy-ip] quit
```

4. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**, configure the device to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.112
[Device-radius-rs1] primary accounting 192.168.0.112
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] user-name-format without-domain
[Device-radius-rs1] quit
[Device] radius session-control enable
```

5. Configure an authentication domain:

# Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[Device] domain dm1
[Device-isp-dm1] authentication portal radius-scheme rs1
[Device-isp-dm1] authorization portal radius-scheme rs1
[Device-isp-dm1] accounting portal radius-scheme rs1
[Device-isp-dm1] quit
[Device] domain default enable dm1
```

6. Configure portal authentication:

# Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable direct portal authentication, specify the portal Web server, and configure the BAS-IP as 2.2.2.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal enable method direct
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
```

```
[Device–GigabitEthernet1/0/2] portal bas-ip 2.2.2.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[Device] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication method: Direct
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: 2.2.2.1
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Portal authentication type: Disabled
     Portal Web server: Not configured
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
```

```
    Pre-auth IP pool: Not configured
    Max portal users: Not configured
    Bas-ipv6: Not configured
    User detection: Not configured
    Portal temp-pass: Disabled
    Action for server detection:
        Server type    Server name                        Action
        --             --                                 --
    Layer3 source network:
        IP address                                        Prefix length

    Destination authenticate subnet:
        IP address                                        Prefix length
    Advertisement push: Not configured
```

A user can perform portal authentication by using the iNode client or a Web browser. Before passing the authentication, the user can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page. After passing the authentication, the user can access other network resources.

# After the user passes authentication, display information about the portal user.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     2.2.2.2            --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: N/A
    Inbound CAR: N/A
    Outbound CAR: N/A
```

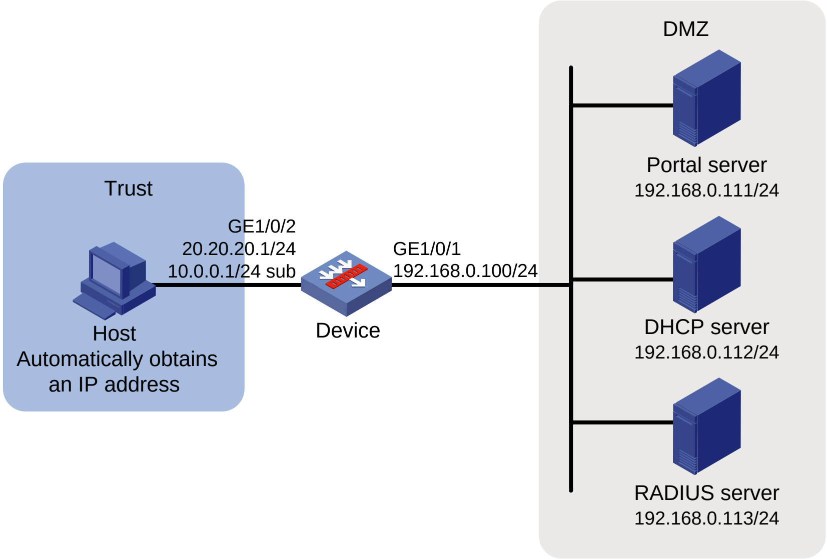
Example: Configuring re-DHCP portal authentication

Network configuration

As shown in Figure 12, the host is directly connected to the device (the access device). The host obtains an IP address through the DHCP server. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure re-DHCP portal authentication. Before passing the authentication, the host is assigned a private IP address. After passing the authentication, the host gets a public IP address and can access network resources.

Figure 12 Network diagram



Prerequisites and guidelines

- For re-DHCP portal authentication, configure a public address pool (20.20.20.0/24) and a private address pool (10.0.0.0/24) on the DHCP server. (Details not shown.)
- For re-DHCP portal authentication:
  - The device must be configured as a DHCP relay agent.
  - The portal-enabled interface must be configured with a primary IP address (a public IP address) and a secondary IP address (a private IP address).

  For information about DHCP relay agent configuration, see *Layer 3—IP Services Configuration Guide*.
- Make sure the IP address of the portal device added on the portal server is the public IP address (20.20.20.1) of the device's interface connecting the host. The private IP address range for the IP address group associated with the portal device is the private subnet 10.0.0.0/24 where the host resides. The public IP address range for the IP address group is the public subnet 20.20.20.0/24.

Procedure

1. Configure the RADIUS server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   ```

```
[Device-security-zone-Trust] quit
```

4. Configure a security policy:

# Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-1-trust-dmz] source-zone trust
[Device-security-policy-ip-1-trust-dmz] destination-zone dmz
[Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-1-trust-dmz] action pass
[Device-security-policy-ip-1-trust-dmz] quit
```

# Configure a rule named **portallocalout1** to allow the device to send packets to the RADIUS server, portal server, and DHCP server.

```
[Device-security-policy-ip] rule name portallocalout1
[Device-security-policy-ip-2-portallocalout1] source-zone local
[Device-security-policy-ip-2-portallocalout1] destination-zone dmz
[Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.111
[Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.112
[Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.113
[Device-security-policy-ip-2-portallocalout1] action pass
[Device-security-policy-ip-2-portallocalout1] quit
```

# Configure a rule named **portallocalin1** to allow the device to receive and process the packets from the RADIUS server, portal server, and DHCP server.

```
[Device-security-policy-ip] rule name portallocalin1
[Device-security-policy-ip-3-portallocalin1] source-zone dmz
[Device-security-policy-ip-3-portallocalin1] destination-zone local
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.113
[Device-security-policy-ip-3-portallocalin1] action pass
[Device-security-policy-ip-3-portallocalin1] quit
```

# Configure a rule named **portallocalin2** to allow the host to send DHCP requests to the DHCP server.

```
[Device-security-policy-ip] rule 4 name portallocalin2
[Device-security-policy-ip-4-portallocalin2] source-zone trust
[Device-security-policy-ip-4-portallocalin2] service dhcp-relay
[Device-security-policy-ip-4-portallocalin2] destination-zone local
[Device-security-policy-ip-4-portallocalin2] action pass
[Device-security-policy-ip-4-portallocalin2] quit
```

# Configure a rule named **portallocalout2** to allow the host to obtain the IP address allocated by the DHCP server.

```
[Device-security-policy-ip] rule 5 name portallocalout2
[Device-security-policy-ip-5-portallocalout2] source-zone local
[Device-security-policy-ip-5-portallocalout2] service dhcp-client
[Device-security-policy-ip-5-portallocalout2] destination-zone trust
[Device-security-policy-ip-5-portallocalout2] action pass
[Device-security-policy-ip-5-portallocalout2] quit
[Device-security-policy-ip] quit
```

5. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**, configure the device to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.113
[Device-radius-rs1] primary accounting 192.168.0.113
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] user-name-format without-domain
[Device-radius-rs1] quit
[Device] radius session-control enable
```

6. Configure an authentication domain:

# Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[Device] domain dm1
[Device-isp-dm1] authentication portal radius-scheme rs1
[Device-isp-dm1] authorization portal radius-scheme rs1
[Device-isp-dm1] accounting portal radius-scheme rs1
[Device-isp-dm1] quit
[Device] domain default enable dm1
```

7. Configure DHCP relay and authorized ARP to allow the host to obtain an IP address from the DHCP server:

```
[Device] dhcp enable
[Device] dhcp relay client-information record
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] ip address 20.20.20.1 255.255.255.0
[Device–GigabitEthernet1/0/2] ip address 10.0.0.1 255.255.255.0 sub
[Device-GigabitEthernet1/0/2] dhcp select relay
[Device-GigabitEthernet1/0/2] dhcp relay server-address 192.168.0.112
[Device-GigabitEthernet1/0/2] arp authorized enable
[Device-GigabitEthernet1/0/2] quit
```

8. Configure portal authentication:

# Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable re-DHCP portal authentication, specify the portal Web server, and configure the BAS-IP as 20.20.20.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] portal enable method redhcp
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 20.20.20.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[Device] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication type: Redhcp
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: 20.20.20.1
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Portal authentication type: Disabled
     Portal Web server: Not configured
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
```

```
    Bas-ipv6: Not configured
    User detection: Not configured
    Portal temp-pass: Disabled
    Action for server detection:
        Server type    Server name                        Action
        --             --                                 --
    Layer3 source network:
        IP address                                        Prefix length

    Destination authenticate subnet:
        IP address                                        Prefix length
    Advertisement push: Not configured
```

Before passing the authentication, a user that uses the iNode client can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page. After passing the authentication, the user can access other network resources.

# After the user passes authentication, display information about the portal user.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     20.20.20.2         --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: N/A
    Inbound CAR: N/A
    Outbound CAR: N/A
```

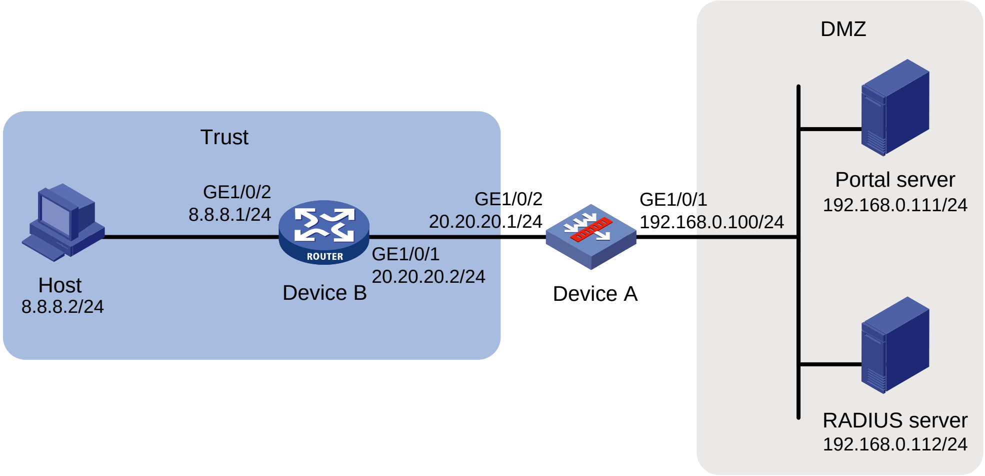
Example: Configuring cross-subnet portal authentication

Network configuration

As shown in Figure 13, Device A supports portal authentication. The host accesses Device A through Device B. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure Device A for cross-subnet portal authentication. Before passing the authentication, the host can access only the portal server. After passing the authentication, the user can access other network resources.

Figure 13 Network diagram



Prerequisites and guidelines

Make sure the IP address of the portal device added on the portal authentication server is the IP address (20.20.20.1) of the device's interface connecting the host. The IP address group associated with the portal device is the subnet of the host (8.8.8.0/24).

Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name dmz
   [DeviceA-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-DMZ] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   ```
4. Configure settings for routing:

   This example configures a static route, and the next hop in the route is 20.20.20.2.

   ```
   [DeviceA] ip route-static 8.8.8.0 24 20.20.20.2
   ```
5. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow the host to access the portal server.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule name trust-dmz
   [DeviceA-security-policy-ip-1-trust-dmz] source-zone trust
   [DeviceA-security-policy-ip-1-trust-dmz] destination-zone dmz
   [DeviceA-security-policy-ip-1-trust-dmz] source-ip-host 8.8.8.2
   [DeviceA-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
   ```

```
[DeviceA-security-policy-ip-1-trust-dmz] action pass
[DeviceA-security-policy-ip-1-trust-dmz] quit
```

# Configure a rule named **portallocalout** to allow Device A to send packets to the RADIUS server and portal server.

```
[DeviceA-security-policy-ip] rule name portallocalout
[DeviceA-security-policy-ip-2-portallocalout] source-zone local
[DeviceA-security-policy-ip-2-portallocalout] destination-zone dmz
[DeviceA-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[DeviceA-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[DeviceA-security-policy-ip-2-portallocalout] action pass
[DeviceA-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow Device A to receive and process the packets from the RADIUS server and portal server.

```
[DeviceA-security-policy-ip] rule name portallocalin
[DeviceA-security-policy-ip-3-portallocalin] source-zone dmz
[DeviceA-security-policy-ip-3-portallocalin] destination-zone local
[DeviceA-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[DeviceA-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[DeviceA-security-policy-ip-3-portallocalin] action pass
[DeviceA-security-policy-ip-3-portallocalin] quit
[DeviceA-security-policy-ip] quit
```

6. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**, configure Device A to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

```
<DeviceA> system-view
[DeviceA] radius scheme rs1
[DeviceA-radius-rs1] primary authentication 192.168.0.112
[DeviceA-radius-rs1] primary accounting 192.168.0.112
[DeviceA-radius-rs1] key authentication simple radius
[DeviceA-radius-rs1] key accounting simple radius
[DeviceA-radius-rs1] user-name-format without-domain
[DeviceA-radius-rs1] quit
[DeviceA] radius session-control enable
```

7. Configure an authentication domain:

# Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[DeviceA] domain dm1
[DeviceA-isp-dm1] authentication portal radius-scheme rs1
[DeviceA-isp-dm1] authorization portal radius-scheme rs1
[DeviceA-isp-dm1] accounting portal radius-scheme rs1
[DeviceA-isp-dm1] quit
[DeviceA] domain default enable dm1
```

8. Configure portal authentication:

# Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable cross-subnet portal authentication, specify the portal Web server, and configure the BAS-IP as 20.20.20.1.

```
[DeviceA] portal server newpt
[DeviceA-portal-server-newpt] ip 192.168.0.111 key simple portal
[DeviceA-portal-server-newpt] port 50100
```

```
[DeviceA-portal-server-newpt] quit
[DeviceA] portal web-server newpt
[DeviceA-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[DeviceA-portal-websvr-newpt] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA–GigabitEthernet1/0/2] portal enable method layer3
[DeviceA–GigabitEthernet1/0/2] portal apply web-server newpt
[DeviceA–GigabitEthernet1/0/2] portal bas-ip 20.20.20.1
[DeviceA–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[DeviceA] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication type: Layer3
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: 20.20.20.1
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Portal authentication type: Disabled
```

```
    Portal Web server: Not configured
    Secondary portal Web server: Not configured
    Portal mac-trigger-server: Not configured
    Authentication domain: Not configured
    Pre-auth domain: Not configured
    Extend-auth domain: Not configured
    User-dhcp-only: Disabled
    Pre-auth IP pool: Not configured
    Max portal users: Not configured
    Bas-ipv6: Not configured
    User detection: Not configured
    Portal temp-pass: Disabled
    Action for server detection:
        Server type    Server name                        Action
        --             --                                 --
    Layer3 source network:
        IP address                                        Prefix length

    Destination authenticate subnet:
        IP address                                        Prefix length
    Advertisement push: Not configured
```

A user can perform portal authentication by using the iNode client or a Web browser. Before passing the authentication, the user can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page. After passing the authentication, the user can access other network resources.

# After the user passes authentication, display information about the portal user.

```
[DeviceA] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     8.8.8.2            --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: N/A
    Inbound CAR: N/A
    Outbound CAR: N/A
```

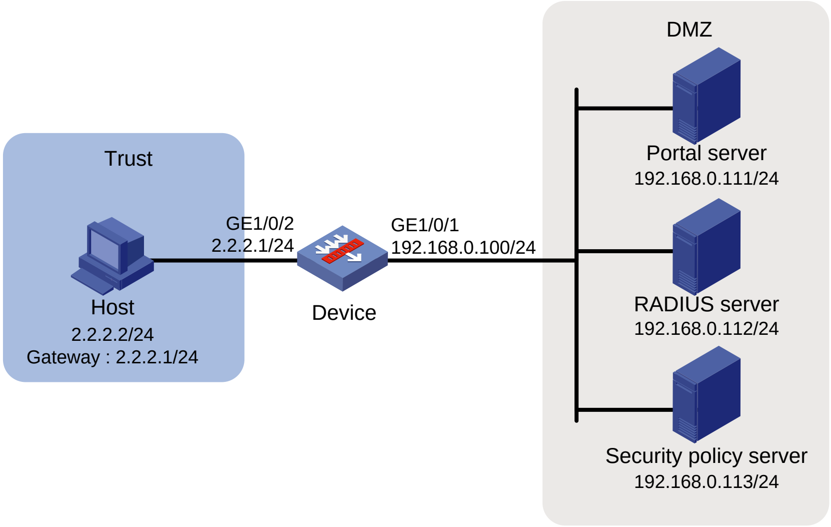
Example: Configuring extended direct portal authentication

Network configuration

As shown in Figure 14, the host is directly connected to the device (the access device). The host is assigned a public IP address either manually or through DHCP. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure extended direct portal authentication. If the host fails security check after passing identity authentication, it can access only subnet 192.168.0.0/24. After passing security check, the host can access other network resources.

Figure 14 Network diagram



Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```
4. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow the host to access the portal server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-dmz
   [Device-security-policy-ip-1-trust-dmz] source-zone trust
   [Device-security-policy-ip-1-trust-dmz] destination-zone dmz
   [Device-security-policy-ip-1-trust-dmz] source-ip-host 2.2.2.2
   [Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
   [Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.113
   [Device-security-policy-ip-1-trust-dmz] action pass
   [Device-security-policy-ip-1-trust-dmz] quit
   ```

# Configure a rule named **portallocalout** to allow the device to send packets to the RADIUS server, portal server, and security policy server.

```
[Device-security-policy-ip] rule name portallocalout
[Device-security-policy-ip-2-portallocalout] source-zone local
[Device-security-policy-ip-2-portallocalout] destination-zone dmz
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.113
[Device-security-policy-ip-2-portallocalout] action pass
[Device-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow the device to receive and process the packets from the RADIUS server, portal server, and security policy server.

```
[Device-security-policy-ip] rule name portallocalin
[Device-security-policy-ip-3-portallocalin] source-zone dmz
[Device-security-policy-ip-3-portallocalin] destination-zone local
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.113
[Device-security-policy-ip-3-portallocalin] action pass
[Device-security-policy-ip-3-portallocalin] quit
[Device-security-policy-ip] quit
```

5. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**. and configure the device to exclude the ISP domain name from the username sent to the RADIUS server.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.112
[Device-radius-rs1] primary accounting 192.168.0.112
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] user-name-format without-domain
```

6. Enable RADIUS session control, and specify a session-control client and shared key.

```
[Device] radius session-control enable
[Device] radius session-control client ip 192.168.0.112 key simple 12345
```

7. Configure an authentication domain:

# Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[Device] domain dm1
[Device-isp-dm1] authentication portal radius-scheme rs1
[Device-isp-dm1] authorization portal radius-scheme rs1
[Device-isp-dm1] accounting portal radius-scheme rs1
[Device-isp-dm1] quit
[Device] domain default enable dm1
```

8. Configure ACL 3000 as the isolation ACL and ACL 3001 as the security ACL.

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip destination 192.168.0.0 0.0.0.255
[Device-acl-ipv4-adv-3000] rule deny ip
[Device-acl-ipv4-adv-3000] quit
```

```
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3001] rule permit ip
[Device-acl-ipv4-adv-3001] quit
```

NOTE:

Make sure you specify ACL 3000 as the isolation ACL and ACL 3001 as the security ACL on the security policy server.

9. Configure portal authentication:

   # Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable direct portal authentication, specify the portal Web server, and configure the BAS-IP as 2.2.2.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal enable method direct
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 2.2.2.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[Device] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication type: Direct
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: 2.2.2.1
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Portal authentication type: Disabled
     Portal Web server: Not configured
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ipv6: Not configured
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address                                        Prefix length

     Destination authenticate subnet:
         IP address                                        Prefix length
     Advertisement push: Not configured
```

Before passing portal authentication, a user that uses the iNode client can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page.

- The user can access the resources permitted by ACL 3000 after passing only identity authentication.
- The user can access network resources permitted by ACL 3001 after passing both identity authentication and security check.

# After the user passes identity authentication and security check, display information about the portal user.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
```

```
VPN instance: N/A
MAC                IP                VLAN   Interface
0015-e9a6-7cfe     2.2.2.2           --     GigabitEthernet1/0/2
Authorization information:
  DHCP IP pool: N/A
  ACL number/name: 3001
  Inbound CAR: N/A
  Outbound CAR: N/A
```

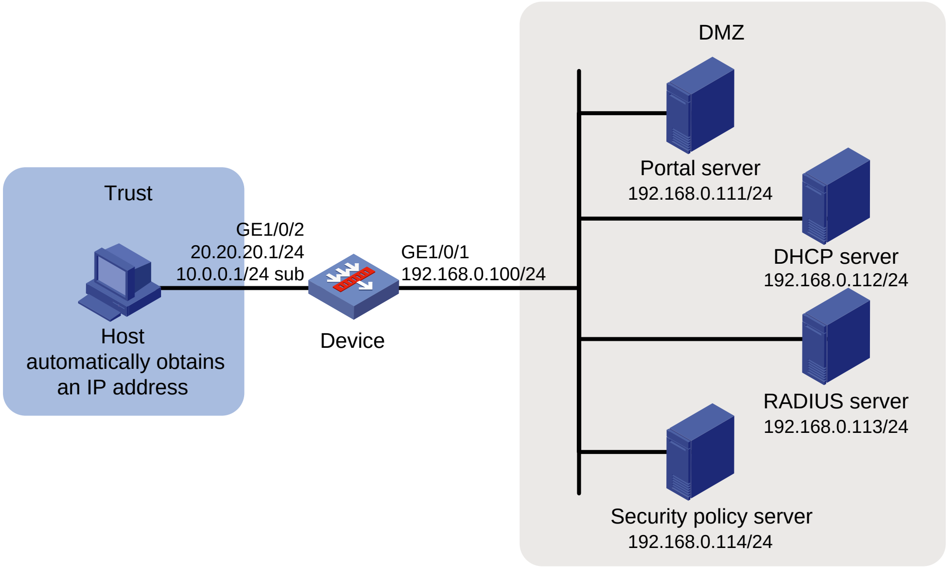
Example: Configuring extended re-DHCP portal authentication

Network configuration

As shown in Figure 15, the host is directly connected to the device (the access device). The host obtains an IP address through the DHCP server. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure extended re-DHCP portal authentication. Before passing portal authentication, the host is assigned a private IP address. After passing portal identity authentication, the host obtains a public IP address and accepts security check. If the host fails the security check, it can access only subnet 192.168.0.0/24. After passing the security check, the host can access other network resources.

Figure 15 Network diagram



Prerequisites and guidelines

- For re-DHCP portal authentication, configure a public address pool (20.20.20.0/24) and a private address pool (10.0.0.0/24) on the DHCP server. (Details not shown.)
- For re-DHCP portal authentication:
  - The device must be configured as a DHCP relay agent.
  - The portal-enabled interface must be configured with a primary IP address (a public IP address) and a secondary IP address (a private IP address).

  For information about DHCP relay agent configuration, see *Layer 3—IP Services Configuration Guide*.

- Make sure the IP address of the portal device added on the portal server is the public IP address (20.20.20.1) of the device's interface connecting the host. The private IP address range for the IP address group associated with the portal device is the private subnet 10.0.0.0/24 where the host resides. The public IP address range for the IP address group is the public subnet 20.20.20.0/24.

Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```
4. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow the host to access the DHCP server, portal server, and security policy server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-dmz
   [Device-security-policy-ip-1-trust-dmz] source-zone trust
   [Device-security-policy-ip-1-trust-dmz] destination-zone dmz
   [Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
   [Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.112
   [Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.114
   [Device-security-policy-ip-1-trust-dmz] action pass
   [Device-security-policy-ip-1-trust-dmz] quit
   ```

   # Configure a rule named **portallocalout1** to allow the device to send packets to the RADIUS server, portal server, DHCP server, and security policy server.

   ```
   [Device-security-policy-ip] rule name portallocalout1
   [Device-security-policy-ip-2-portallocalout1] source-zone local
   [Device-security-policy-ip-2-portallocalout1] destination-zone dmz
   [Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.111
   [Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.112
   [Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.113
   [Device-security-policy-ip-2-portallocalout1] destination-ip-host 192.168.0.114
   [Device-security-policy-ip-2-portallocalout1] action pass
   [Device-security-policy-ip-2-portallocalout1] quit
   ```

   # Configure a rule named **portallocalin1** to allow the device to receive and process the packets from the RADIUS server, portal server, DHCP server, and security policy server.

   ```
   [Device-security-policy-ip] rule name portallocalin1
   ```

```
[Device-security-policy-ip-3-portallocalin1] source-zone dmz
[Device-security-policy-ip-3-portallocalin1] destination-zone local
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.113
[Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.114
[Device-security-policy-ip-3-portallocalin1] action pass
[Device-security-policy-ip-3-portallocalin1] quit
```

# Configure a rule named **portallocalin2** to allow the host to send DHCP requests to the device.

```
[Device-security-policy-ip] rule name portallocalin2
[Device-security-policy-ip-4-portallocalin2] source-zone trust
[Device-security-policy-ip-4-portallocalin2] service dhcp-relay
[Device-security-policy-ip-4-portallocalin2] destination-zone local
[Device-security-policy-ip-4-portallocalin2] action pass
[Device-security-policy-ip-4-portallocalin2] quit
```

# Configure a rule named **portallocalout2** to allow the host to obtain the IP address allocated by the DHCP server.

```
[Device-security-policy-ip] rule name portallocalout2
[Device-security-policy-ip-5-portallocalout2] source-zone local
[Device-security-policy-ip-4-portallocalout2] service dhcp-client
[Device-security-policy-ip-5-portallocalout2] destination-zone trust
[Device-security-policy-ip-5-portallocalout2] action pass
[Device-security-policy-ip-5-portallocalout2] quit
[Device-security-policy-ip] quit
```

5. Configure a RADIUS scheme:

   # Create a RADIUS scheme named **rs1**, and configure the device to exclude the ISP domain name from the username sent to the RADIUS server.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.113
[Device-radius-rs1] primary accounting 192.168.0.113
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] user-name-format without-domain
```

6. Enable RADIUS session control, and specify a session-control client and shared key.

```
[Device] radius session-control enable
[Device] radius session-control client ip 192.168.0.113 key simple 12345
```

7. Configure an authentication domain:

   # Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[Device] domain dm1
[Device-isp-dm1] authentication portal radius-scheme rs1
[Device-isp-dm1] authorization portal radius-scheme rs1
[Device-isp-dm1] accounting portal radius-scheme rs1
[Device-isp-dm1] quit
[Device] domain default enable dm1
```

8. Configure ACL 3000 as the isolation ACL and ACL 3001 as the security ACL.

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip destination 192.168.0.0 0.0.0.255
[Device-acl-ipv4-adv-3000] rule deny ip
[Device-acl-ipv4-adv-3000] quit
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3001] rule permit ip
[Device-acl-ipv4-adv-3001] quit
```

NOTE:

Make sure you specify ACL 3000 as the isolation ACL and ACL 3001 as the security ACL on the security policy server.

9. Configure DHCP relay and authorized ARP:

```
[Device] dhcp enable
[Device] dhcp relay client-information record
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] ip address 20.20.20.1 255.255.255.0
[Device–GigabitEthernet1/0/2] ip address 10.0.0.1 255.255.255.0 sub
[Device-GigabitEthernet1/0/2] dhcp select relay
[Device-GigabitEthernet1/0/2] dhcp relay server-address 192.168.0.112
[Device-GigabitEthernet1/0/2] arp authorized enable
[Device-GigabitEthernet1/0/2] quit
```

10. Configure portal authentication:

# Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable re-DHCP portal authentication, specify the portal Web server, and configure the BAS-IP as 20.20.20.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] portal enable method redhcp
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 20.20.20.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[Device] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
```

```
    Advertisement-push     : Disabled
    Embedded advertisement : Disabled
IPv4:
    Portal status: Enabled
    Portal authentication type: Redhcp
    Portal Web server: newpt(active)
    Secondary portal Web server: Not configured
    Portal mac-trigger-server: Not configured
    Authentication domain: Not configured
    Pre-auth domain: Not configured
    Extend-auth domain: Not configured
    User-dhcp-only: Disabled
    Pre-auth IP pool: Not configured
    Max portal users: Not configured
    Bas-ip: 20.20.20.1
    User detection: Not configured
    Portal temp-pass: Disabled
    Action for server detection:
        Server type    Server name                        Action
        --             --                                 --
    Layer3 source network:
        IP address               Mask

    Destination authenticate subnet:
        IP address               Mask
    Advertisement push: Not configured
IPv6:
    Portal status: Disabled
    Portal authentication type: Disabled
    Portal Web server: Not configured
    Secondary portal Web server: Not configured
    Portal mac-trigger-server: Not configured
    Authentication domain: Not configured
    Pre-auth domain: Not configured
    Extend-auth domain: Not configured
    User-dhcp-only: Disabled
    Pre-auth IP pool: Not configured
    Max portal users: Not configured
    Bas-ipv6: Not configured
    User detection: Not configured
    Portal temp-pass: Disabled
    Action for server detection:
        Server type    Server name                        Action
        --             --                                 --
    Layer3 source network:
        IP address                                        Prefix length

    Destination authenticate subnet:
        IP address                                          Prefix length
    Advertisement push: Not configured
```

Before passing portal authentication, a user that uses the iNode client can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page.

- The user can access the resources permitted by ACL 3000 after passing only identity authentication.
- The user can access network resources permitted by ACL 3001 after passing both identity authentication and security check.

# After the user passes identity authentication and security check, display information about the portal user.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     20.20.20.2         --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: 3001
    Inbound CAR: N/A
    Outbound CAR: N/A
```

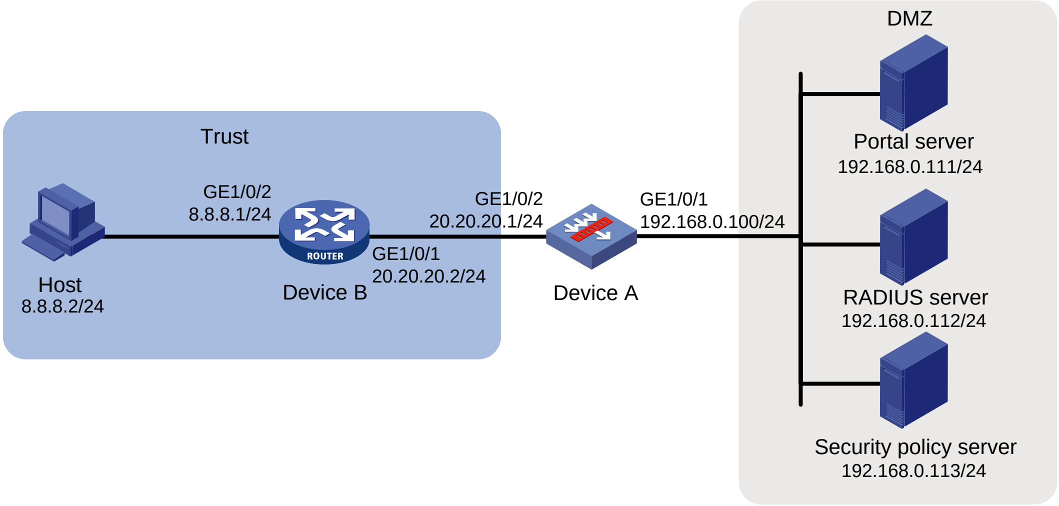
Example: Configuring extended cross-subnet portal authentication

Network configuration

As shown in Figure 16, Device A supports portal authentication. The host accesses Device A through Device B. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure Device A for extended cross-subnet portal authentication. Before passing portal authentication, the host can access only the portal server. After passing portal identity authentication, the host accepts security check. If the host fails the security check it can access only the subnet 192.168.0.0/24. After passing the security check, the host can access other network resources.

Figure 16 Network diagram



Prerequisites and guidelines

Make sure the IP address of the portal device added on the portal server is the IP address (20.20.20.1) of the device's interface connecting the host. The IP address group associated with the portal device is the subnet of the host (8.8.8.0/24).

Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name dmz
   [DeviceA-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-DMZ] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   ```
4. Configure settings for routing:

   This example configures a static route, and the next hop in the route is 20.20.20.2.

   ```
   [DeviceA] ip route-static 8.8.8.0 24 20.20.20.2
   ```
5. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow the host to access the RADIUS server, portal server, and security policy server.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule name trust-dmz
   [DeviceA-security-policy-ip-1-trust-dmz] source-zone trust
   [DeviceA-security-policy-ip-1-trust-dmz] destination-zone dmz
   ```

```
[DeviceA-security-policy-ip-1-trust-dmz] source-ip-host 8.8.8.2
[DeviceA-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
[DeviceA-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.112
[DeviceA-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.113
[DeviceA-security-policy-ip-1-trust-dmz] action pass
[DeviceA-security-policy-ip-1-trust-dmz] quit
```

# Configure a rule named **portallocalout** to allow Device A to send packets to the RADIUS server, portal server, and security policy server.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name portallocalout
[DeviceA-security-policy-ip-2-portallocalout] source-zone local
[DeviceA-security-policy-ip-2-portallocalout] destination-zone dmz
[DeviceA-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[DeviceA-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[DeviceA-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.113
[DeviceA-security-policy-ip-2-portallocalout] action pass
[DeviceA-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow Device A to receive and process the packets from the RADIUS server, portal server, and security policy server.

```
[DeviceA-security-policy-ip] rule name portallocalin
[DeviceA-security-policy-ip-3-portallocalin] source-zone dmz
[DeviceA-security-policy-ip-3-portallocalin] destination-zone local
[DeviceA-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[DeviceA-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[DeviceA-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.113
[DeviceA-security-policy-ip-3-portallocalin] action pass
[DeviceA-security-policy-ip-3-portallocalin] quit
[Device-security-policy-ip] quit
```

6. Configure a RADIUS scheme:

   # Create a RADIUS scheme named **rs1**, and configure Device A to exclude the ISP domain name from the username sent to the RADIUS server.

```
<DeviceA> system-view
[DeviceA] radius scheme rs1
[DeviceA-radius-rs1] primary authentication 192.168.0.112
[DeviceA-radius-rs1] primary accounting 192.168.0.112
[DeviceA-radius-rs1] key authentication simple radius
[DeviceA-radius-rs1] key accounting simple radius
[DeviceA-radius-rs1] user-name-format without-domain
```

7. Enable RADIUS session control, and specify a session-control client and shared key.

```
[DeviceA] radius session-control enable
[DeviceA] radius session-control client ip 192.168.0.112 key simple 12345
```

8. Configure an authentication domain:

   # Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[DeviceA] domain dm1
[DeviceA-isp-dm1] authentication portal radius-scheme rs1
[DeviceA-isp-dm1] authorization portal radius-scheme rs1
[DeviceA-isp-dm1] accounting portal radius-scheme rs1
```

```
[DeviceA-isp-dm1] quit
[DeviceA] domain default enable dm1
```

9. Configure ACL 3000 as the isolation ACL and ACL 3001 as the security ACL.

```
[DeviceA] acl advanced 3000
[DeviceA-acl-ipv4-adv-3000] rule permit ip destination 192.168.0.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3000] rule deny ip
[DeviceA-acl-ipv4-adv-3000] quit
[DeviceA] acl advanced 3001
[DeviceA-acl-ipv4-adv-3001] rule permit ip
[DeviceA-acl-ipv4-adv-3001] quit
```

NOTE:

Make sure you specify ACL 3000 as the isolation ACL and ACL 3001 as the security ACL on the security policy server.

10. Configure portal authentication:

# Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable cross-subnet portal authentication, specify the portal Web server newpt, and configure the BAS-IP as 20.20.20.1.

```
[DeviceA] portal server newpt
[DeviceA-portal-server-newpt] ip 192.168.0.111 key simple portal
[DeviceA-portal-server-newpt] port 50100
[DeviceA-portal-server-newpt] quit
[Device] portal web-server newpt
[DeviceA-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[DeviceA-portal-websvr-newpt] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA–GigabitEthernet1/0/2] portal enable method layer3
[DeviceA–GigabitEthernet1/0/2] portal apply web-server newpt
[DeviceA–GigabitEthernet1/0/2] portal bas-ip 20.20.20.1
[DeviceA–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[DeviceA] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication type: Layer3
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: 20.20.20.1
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Authentication type: Disabled
     Portal Web server: Not configured
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ipv6: Not configured
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address                                        Prefix length

     Destination authenticate subnet:
         IP address                                        Prefix length
     Advertisement push: Not configured
```

Before passing portal authentication, a user that uses the iNode client can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user are redirected to the authentication page.

- The user can access the resources permitted by ACL 3000 after passing only identity authentication.

- The user can access network resources permitted by ACL 3001 after passing both identity authentication and security check.

# After the user passes identity authentication and security check, display information about the portal user.

```
[DeviceA] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     8.8.8.2            --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: 3001
    Inbound CAR: N/A
    Outbound CAR: N/A
```

Example: Configuring portal server detection and portal user synchronization

Network configuration

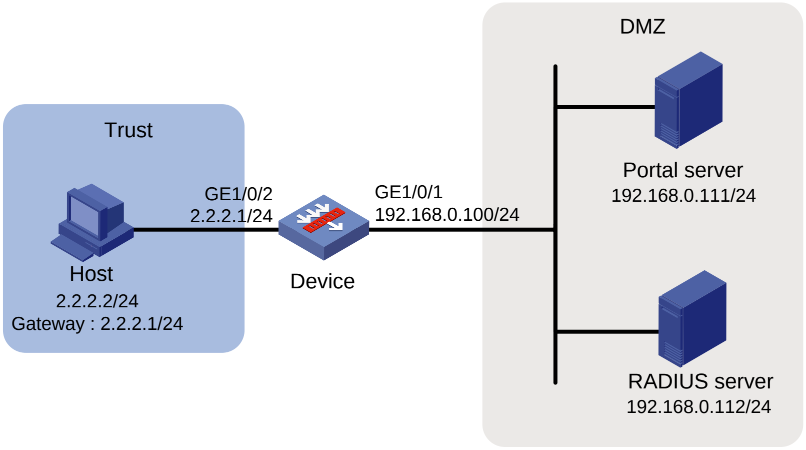
As shown in Figure 17, the host is directly connected to the device (the access device). The host is assigned a public IP address either manually or through DHCP. An IMC server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server. In this example, the IMC server runs IMC 7.1 (E0303) and IMC UAM 7.1 (E0304).

Configure direct portal authentication on the device, so the host can access only the portal server before passing the authentication and access other network resources after passing the authentication.

Configure the device to do the following:

- Detect the reachability state of the portal authentication server.
- Send log messages upon state changes.
- Disable portal authentication when the authentication server is unreachable.
- Synchronize portal user information with the portal server periodically.

Figure 17 Network diagram



Prerequisites and guidelines

- Configure the portal authentication server. Be sure to enable the server heartbeat function and the user heartbeat function.
- Configure the device (access device) as follows:
  - Configure direct portal authentication on GigabitEthernet 1/0/2, the interface to which the host is connected.
  - Configure portal authentication server detection, so that the device can detect the reachability of the portal authentication server by cooperating with the portal server heartbeat function.
  - Configure portal user synchronization, so that the device can synchronize portal user information with the portal authentication server by cooperating with the portal user heartbeat function.

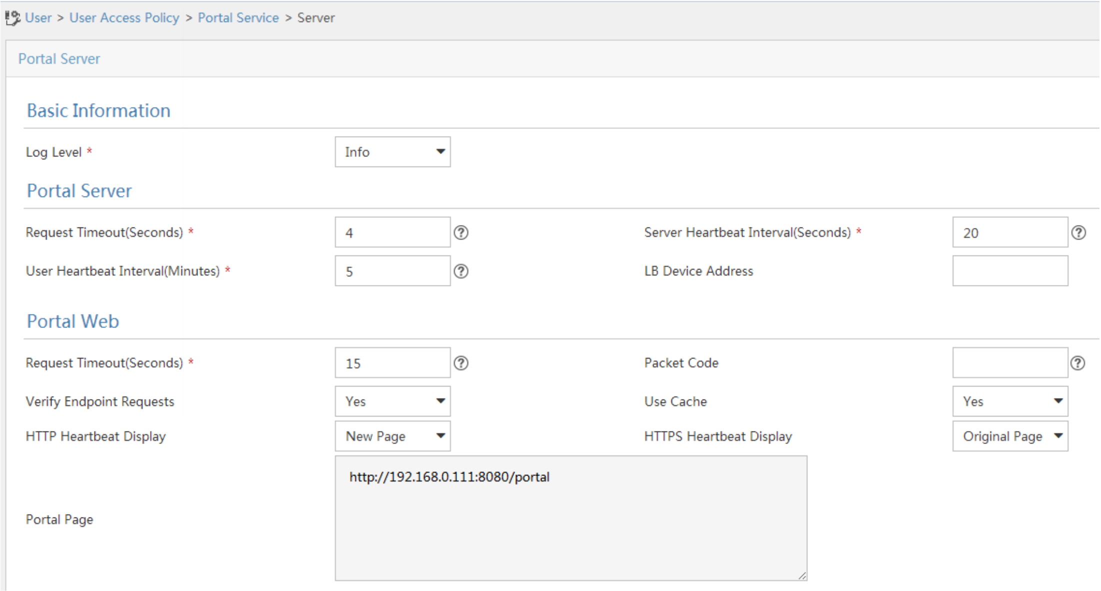
Configuring the RADIUS server

# Configure the RADIUS server correctly to provide authentication and accounting functions. (Details not shown.)

Configuring the portal server

1. Configure the portal server:
   a. Log in to IMC and click the **Service** tab.
   b. Select **User Access Policy** > **Portal Service** > **Server** from the navigation tree to open the portal server configuration page as shown in Figure 18.
   c. Configure the portal server parameter as needed.
      This example uses the default settings.
   d. Click **OK**.

Figure 18 Portal authentication server configuration



User > User Access Policy > Portal Service > Server

Portal Server

Basic Information

Log Level *	Info

Portal Server

Request Timeout(Seconds) *	4	Server Heartbeat Interval(Seconds) *	20
User Heartbeat Interval(Minutes) *	5	LB Device Address	

Portal Web

Request Timeout(Seconds) *	15	Packet Code	
Verify Endpoint Requests	Yes	Use Cache	Yes
HTTP Heartbeat Display	New Page	HTTPS Heartbeat Display	Original Page
Portal Page	http://192.168.0.111:8080/portal		

2. Configure the IP address group:

   a. Select **User Access Policy** > **Portal Service** > **IP Group** from the navigation tree to open the portal IP address group configuration page.

   b. Click **Add** to open the page as shown in Figure 19.

   c. Enter the IP group name.

   d. Enter the start IP address and end IP address of the IP group.

      Make sure the host IP address is in the IP group.

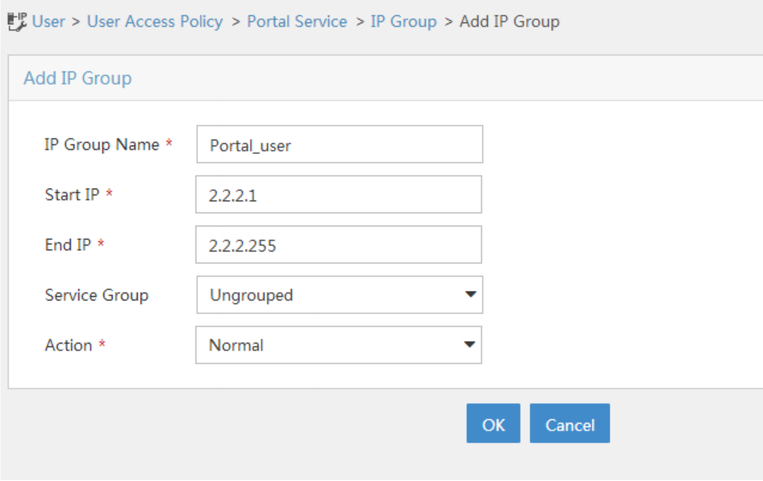
   e. Select a service group.

      This example uses the default group **Ungrouped**.

   f. Select **Normal** from the **Action** list.

   g. Click **OK**.

Figure 19 Adding an IP address group



User > User Access Policy > Portal Service > IP Group > Add IP Group

Add IP Group

IP Group Name *	Portal_user
Start IP *	2.2.2.1
End IP *	2.2.2.255
Service Group	Ungrouped
Action *	Normal

3. Add a portal device:

   a. Select **User Access Policy** > **Portal Service** > **Device** from the navigation tree to open the portal device configuration page.

b. Click **Add** to open the page as shown in Figure 20.

c. Enter device name **NAS**.

d. Enter the IP address of the device's interface connected to the host.

e. Select whether to support sever heartbeat and user heartbeat functions.

   In this example, select **Yes** for both **Support Server Heartbeat** and **Support User Heartbeat**.

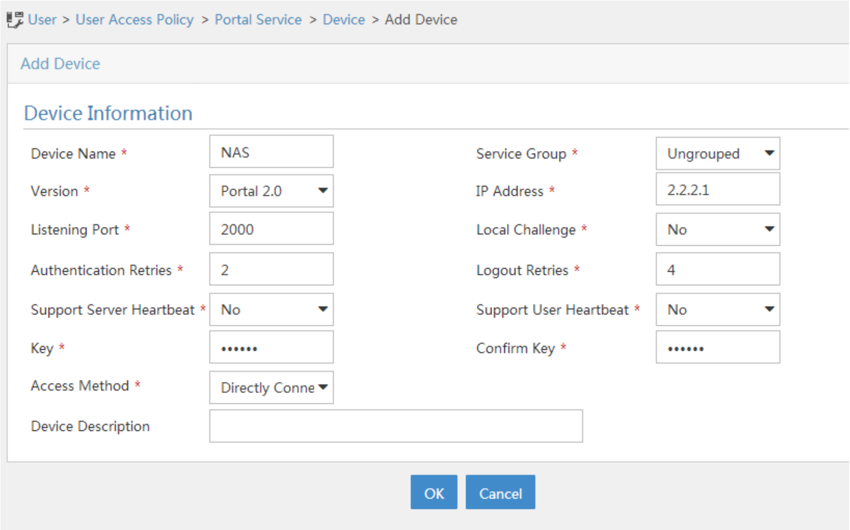
f. Enter the key, which must be the same as that configured on the device.

g. Select **Directly Connected** from the **Access Method** list.

h. Use the default settings for other parameters.

i. Click **OK**.

Figure 20 Adding a portal device



User > User Access Policy > Portal Service > Device > Add Device

Add Device

Device Information

Device Name *	NAS	Service Group *	Ungrouped
Version *	Portal 2.0	IP Address *	2.2.2.1
Listening Port *	2000	Local Challenge *	No
Authentication Retries *	2	Logout Retries *	4
Support Server Heartbeat *	No	Support User Heartbeat *	No
Key *	••••••	Confirm Key *	••••••
Access Method *	Directly Conne		
Device Description			

4. Associate the portal device with the IP address group:

   a. As shown in Figure 21, click the **Port Group Information Management** icon for device **NAS** to open the port group configuration page.

   b. Click **Add** to open the page as shown in Figure 22.

   c. Enter the port group name.

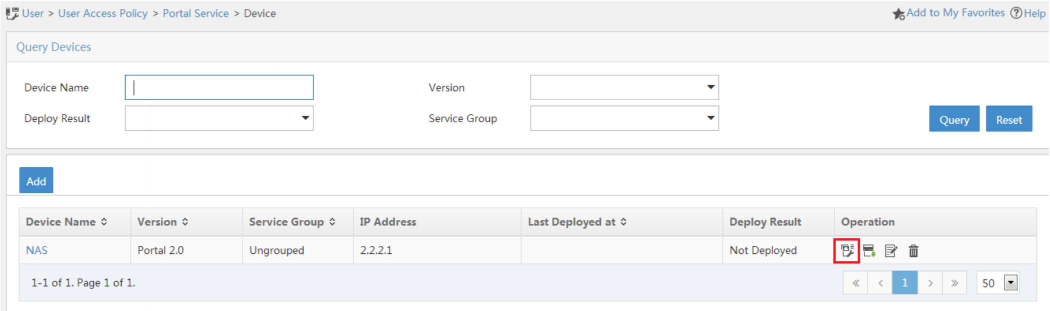
   d. Select the configured IP address group.

      The IP address used by the user to access the network must be within this IP address group.

   e. Use the default settings for other parameters.

   f. Click **OK**.

Figure 21 Device list



User > User Access Policy > Portal Service > Device

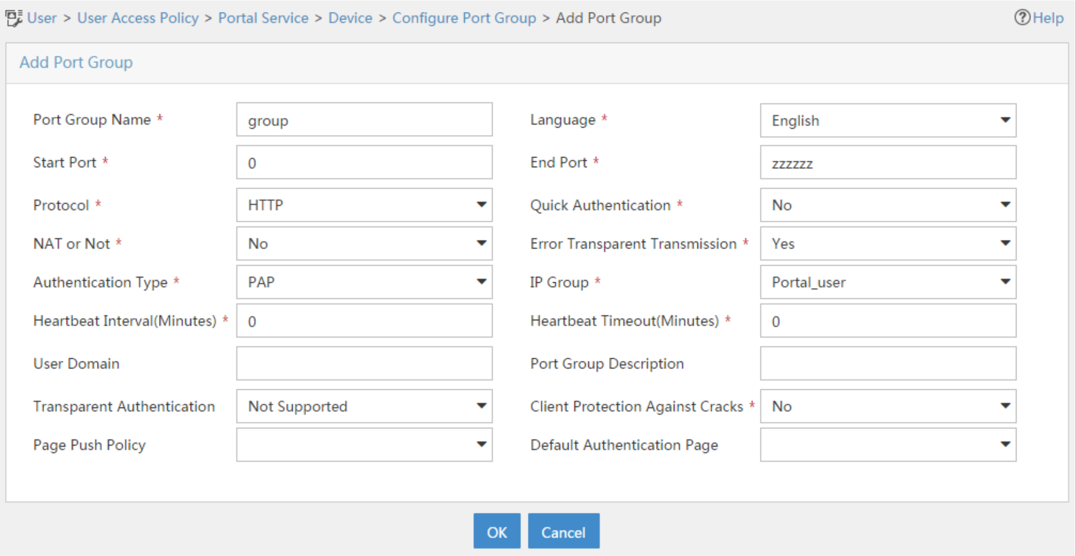
Query Devices

Device Name		Version	
Deploy Result		Service Group	

Device Name	Version	Service Group	IP Address	Last Deployed at	Deploy Result	Operation
NAS	Portal 2.0	Ungrouped	2.2.2.1		Not Deployed	

1-1 of 1. Page 1 of 1.

Figure 22 Adding a port group



User > User Access Policy > Portal Service > Device > Configure Port Group > Add Port Group

Add Port Group

Port Group Name *	group	Language *	English
Start Port *	0	End Port *	zzzzzz
Protocol *	HTTP	Quick Authentication *	No
NAT or Not *	No	Error Transparent Transmission *	Yes
Authentication Type *	PAP	IP Group *	Portal_user
Heartbeat Interval(Minutes) *	0	Heartbeat Timeout(Minutes) *	0
User Domain		Port Group Description	
Transparent Authentication	Not Supported	Client Protection Against Cracks *	No
Page Push Policy		Default Authentication Page	

Configuring the device

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
[Device-security-zone-DMZ] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-1-trust-dmz] source-zone trust
[Device-security-policy-ip-1-trust-dmz] destination-zone dmz
[Device-security-policy-ip-1-trust-dmz] source-ip-host 2.2.2.2
[Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-1-trust-dmz] action pass
[Device-security-policy-ip-1-trust-dmz] quit
```

# Configure a rule named **portallocalout** to allow the device to send packets to the RADIUS server and portal server

```
[Device-security-policy-ip] rule name portallocalout
[Device-security-policy-ip-2-portallocalout] source-zone local
```

```
[Device-security-policy-ip-2-portallocalout] destination-zone dmz
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[Device-security-policy-ip-2-portallocalout] action pass
[Device-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow the device to receive and process the packets from the RADIUS server and portal server.

```
[Device-security-policy-ip] rule name portallocalin
[Device-security-policy-ip-3-portallocalin] source-zone dmz
[Device-security-policy-ip-3-portallocalin] destination-zone local
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin] action pass
[Device-security-policy-ip-3-portallocalin] quit
[Device-security-policy-ip] quit
```

4. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**, configure the device to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.112
[Device-radius-rs1] primary accounting 192.168.0.112
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] user-name-format without-domain
[Device-radius-rs1] quit
[Device] radius session-control enable
```

5. Configure an authentication domain:

# Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

```
[Device] domain dm1
[Device-isp-dm1] authentication portal radius-scheme rs1
[Device-isp-dm1] authorization portal radius-scheme rs1
[Device-isp-dm1] accounting portal radius-scheme rs1
[Device-isp-dm1] quit
[Device] domain default enable dm1
```

6. Configure portal authentication:

# Configure a portal authentication server, and configure reachability detection of the portal authentication server: set the server detection interval and specify the action upon reachability status changes.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] server-detect timeout 40 log
```

NOTE:

The value of **timeout** must be greater than or equal to the portal server heartbeat interval.

# Configure portal user synchronization with the portal authentication server, and set the synchronization detection interval to 600 seconds.

```
[Device-portal-server-newpt] user-sync timeout 600
[Device-portal-server-newpt] quit
```

NOTE:

The value of **timeout** must be greater than or equal to the portal user heartbeat interval.

# Configure a portal Web server. On GigabitEthernet 1/0/2, enable direct portal authentication, enable portal fail-permit for the portal authentication server, specify the portal Web server, and configure the BAS-IP as 2.2.2.1.

```
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal enable method direct
[Device–GigabitEthernet1/0/2] portal fail-permit server newpt
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 2.2.2.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Display information about the portal authentication server.

```
[Device] display portal server newpt
Portal server: newpt
  Type                     : IMC
  IP                       : 192.168.0.111
  VPN instance             : Not configured
  Port                     : 50100
  Server Detection         : Timeout 40s  Action: log
  User synchronization     : Timeout 600s
  Status                   : Up
  Exclude-attribute        : Not configured
  Logout notification      : Not configured
```

The **Up** status of the portal authentication server indicates that the portal authentication server is reachable. If the access device detects that the portal authentication server is unreachable, the **Status** field in the command output displays **Down**. The access device generates a server unreachable log "Portal server newpt turns down from up." and disables portal authentication on the access interface, so the host can access the external network without authentication.

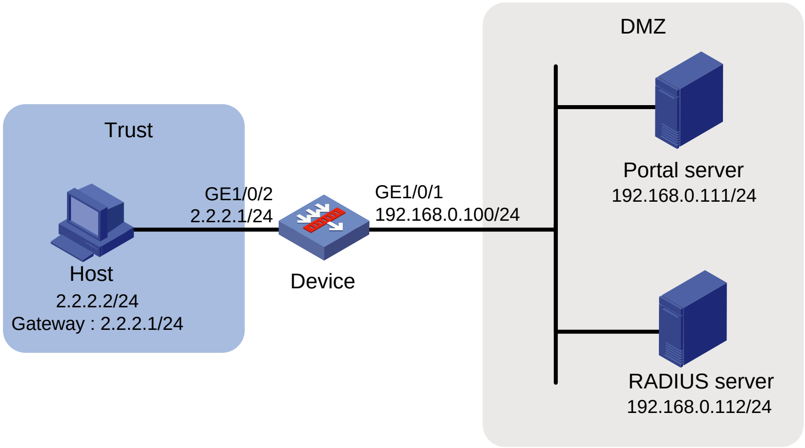
Example: Configuring direct portal authentication with a preauthentication domain

Network configuration

As shown in Figure 23, the host is directly connected to the device (the access device). The host is assigned a public IP address through DHCP. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure direct portal authentication, so the host can access only subnet 192.168.0.0/24 before passing the authentication and access other network resources after passing the authentication.

Figure 23 Network diagram



Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```
4. Configure a security policy:

   # Configure a rule named **portallocalin1** to allow the host to send DHCP requests to the device.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name portallocalin1
   [Device-security-policy-ip-1-portallocalin1] source-zone trust
   [Device-security-policy-ip-1-portallocalin1] service dhcp-server
   [Device-security-policy-ip-1-portallocalin1] destination-zone local
   [Device-security-policy-ip-1-portallocalin1] action pass
   [Device-security-policy-ip-1-portallocalin1] quit
   ```

   # Configure a rule named **portallocalout1** to allow the device to send DHCP response packets to the host.

   ```
   [Device-security-policy-ip] rule name portallocalout1
   [Device-security-policy-ip-2-portallocalout1] source-zone local
   [Device-security-policy-ip-2-portallocalout1] service dhcp-client
   [Device-security-policy-ip-2-portallocalout1] destination-zone trust
   ```

```
[Device-security-policy-ip-2-portallocalout1] action pass
[Device-security-policy-ip-2-portallocalout1] quit
```

# Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-3-trust-dmz] source-zone trust
[Device-security-policy-ip-3-trust-dmz] destination-zone dmz
[Device-security-policy-ip-3-trust-dmz] source-ip-subnet 2.2.2.0 24
[Device-security-policy-ip-3-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-3-trust-dmz] action pass
[Device-security-policy-ip-3-trust-dmz] quit
```

# Configure a rule named **portallocalout2** to allow the device to send packets to the RADIUS server and portal server.

```
[Device-security-policy-ip] rule name portallocalout2
[Device-security-policy-ip-4-portallocalout2] source-zone local
[Device-security-policy-ip-4-portallocalout2] destination-zone dmz
[Device-security-policy-ip-4-portallocalout2] destination-ip-host 192.168.0.111
[Device-security-policy-ip-4-portallocalout2] destination-ip-host 192.168.0.112
[Device-security-policy-ip-4-portallocalout2] action pass
[Device-security-policy-ip-4-portallocalout2] quit
```

# Configure a rule named **portallocalin2** to allow the device to receive and process packets from the RADIUS server and portal server.

```
[Device-security-policy-ip] rule name portallocalin2
[Device-security-policy-ip-5-portallocalin2] source-zone dmz
[Device-security-policy-ip-5-portallocalin2] destination-zone local
[Device-security-policy-ip-5-portallocalin2] source-ip-host 192.168.0.111
[Device-security-policy-ip-5-portallocalin2] source-ip-host 192.168.0.112
[Device-security-policy-ip-5-portallocalin2] action pass
[Device-security-policy-ip-5-portallocalin2] quit
[Device-security-policy-ip] quit
```

5. Configure a preauthentication IP address pool, and enable the DHCP server on GigabitEthernet 1/0/2.

```
<Device> system-view
[Device] dhcp server ip-pool pre
[Device-dhcp-pool-pre] gateway-list 2.2.2.1
[Device-dhcp-pool-pre] network 2.2.2.0 24
[Device-dhcp-pool-pre] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] dhcp select server
[Device–GigabitEthernet1/0/2] quit
```

6. Configure a preauthentication domain:

# Create an ISP domain named **abc**, and specify authorization ACL 3010 in the domain. In ACL 3010, configure a rule to permit the traffic to the subnet 192.168.0.0/24. On GigabitEthernet 1/0/2, configure preauthentication domain as ISP domain **abc**.

```
[Device] domain abc
[Device-isp-abc] authorization-attribute acl 3010
[Device-isp-abc] quit
[Device] acl advanced 3010
[Device-acl-ipv4-adv-3010] rule 1 permit ip destination 192.168.0.0 0.0.0.255
[Device-acl-ipv4-adv-3010] quit
```

```
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal pre-auth domain abc
[Device–GigabitEthernet1/0/2] quit
```

7. Configure portal authentication:

   # Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable direct portal authentication, specify the portal Web server, and configure the BAS-IP as 2.2.2.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal enable method direct
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 2.2.2.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify the portal configuration by executing the **display portal interface** command. (Details not shown.)

# Display information about preauthentication portal users.

```
[Device] display portal user pre-auth interface gigabitethernet 1/0/2
Total portal pre-auth users: 1
MAC               IP                VLAN   Interface
0015-e9a6-7cfe    2.2.2.4           --     GigabitEthernet1/0/2
  State: Online
  VPN instance: --
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: 3010
    Inbound CAR: N/A
    Outbound CAR: N/A
```

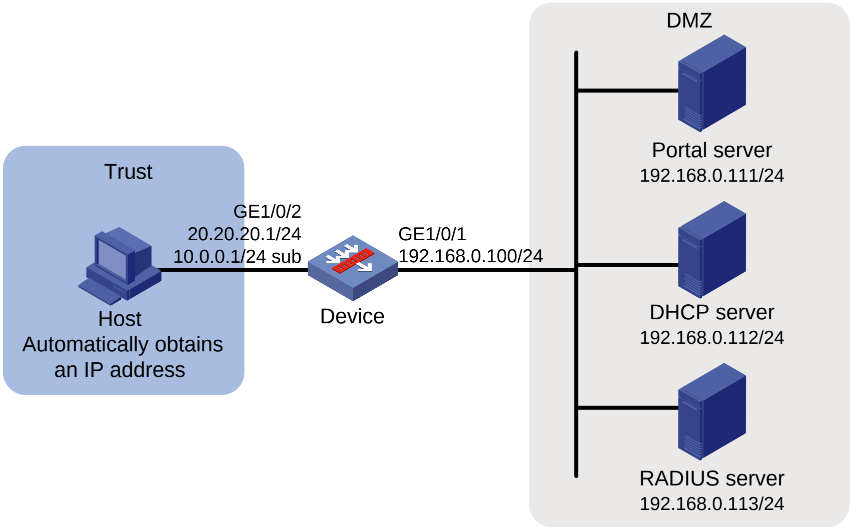
Example: Configuring re-DHCP portal authentication with a preauthentication domain

Network configuration

As shown in Figure 24, the host is directly connected to the device (the access device). The host obtains an IP address through the DHCP server. A portal server acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure re-DHCP portal authentication. Before passing the authentication, the host is assigned a private IP address and can access only the subnet 192.168.0.0/24. After passing the authentication, the host gets a public IP address and can access other network resources.

Figure 24 Network diagram



Prerequisites and guidelines

- For re-DHCP portal authentication, configure a public address pool (20.20.20.0/24) and a private address pool (10.0.0.0/24) on the DHCP server. (Details not shown.)
- For re-DHCP portal authentication:
  - The device must be configured as a DHCP relay agent.
  - The portal-enabled interface must be configured with a primary IP address (a public IP address) and a secondary IP address (a private IP address).

  For information about DHCP relay agent configuration, see *Layer 3—IP Services Configuration Guide*.
- Make sure the IP address of the portal device added on the portal server is the public IP address (20.20.20.1) of the device's interface connecting the host. The private IP address range for the IP address group associated with the portal device is the private subnet 10.0.0.0/24 where the host resides. The public IP address range for the IP address group is the public subnet 20.20.20.0/24.
- If you have configured a preauthentication IP address pool on portal-enabled interfaces, configure a DHCP relay address pool with the same name on the device. For the DHCP relay address pool, specify the subnet address where the unauthenticated users reside (with the **export-router** keyword specified) and the DHCP server address.

Procedure

1. Configure the RADIUS server and the portal server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   ```

```
[Device-security-zone-DMZ] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

4. Configure a security policy:

   # Configure a rule named **portallocalin1** to allow the host to send DHCP requests to the device.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name portallocalin1
[Device-security-policy-ip-1-portallocalin1] source-zone trust
[Device-security-policy-ip-1-portallocalin1] service dhcp-relay
[Device-security-policy-ip-1-portallocalin1] destination-zone local
[Device-security-policy-ip-1-trust-local] action pass
[Device-security-policy-ip-1-trust-local] quit
```

   # Configure a rule named **portallocalout1** to allow the device to send DHCP response packets to the host.

```
[Device-security-policy-ip] rule name portallocalout1
[Device-security-policy-ip-2-portallocalout1] source-zone local
[Device-security-policy-ip-2-portallocalout1] service dhcp-client
[Device-security-policy-ip-2-portallocalout1] destination-zone trust
[Device-security-policy-ip-2-portallocalout1] action pass
[Device-security-policy-ip-2-portallocalout1] quit
```

   # Configure a rule named **portallocalout2** to allow the device to send packets to the RADIUS server, portal server, and DHCP server.

```
[Device-security-policy-ip] rule name portallocalout2
[Device-security-policy-ip-3-portallocalout2] source-zone local
[Device-security-policy-ip-3-portallocalout2] destination-zone dmz
[Device-security-policy-ip-3-portallocalout2] destination-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalout2] destination-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalout2] destination-ip-host 192.168.0.113
[Device-security-policy-ip-3-portallocalout2] action pass
[Device-security-policy-ip-3-portallocalout2] quit
```

   # Configure a rule named **portallocalin2** to allow the device to receive and process the packets from the RADIUS server, portal server, and DHCP server.

```
[Device-security-policy-ip] rule name portallocalin2
[Device-security-policy-ip-4-portallocalin2] source-zone dmz
[Device-security-policy-ip-4-portallocalin2] destination-zone local
[Device-security-policy-ip-4-portallocalin2] source-ip-host 192.168.0.111
[Device-security-policy-ip-4-portallocalin2] source-ip-host 192.168.0.112
[Device-security-policy-ip-4-portallocalin2] source-ip-host 192.168.0.113
[Device-security-policy-ip-4-portallocalin2] action pass
[Device-security-policy-ip-4-portallocalin2] quit
```

   # Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-5-trust-dmz] source-zone trust
[Device-security-policy-ip-5-trust-dmz] destination-zone dmz
[Device-security-policy-ip-5-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-5-trust-dmz] action pass
[Device-security-policy-ip-5-trust-dmz] quit
```

```
[Device-security-policy-ip] quit
```

5. Configure a preauthentication domain:

   # Create an ISP domain named **abc**, and specify authorization ACL 3010 in the domain. In ACL 3010, configure a rule to permit the traffic to the subnet 192.168.0.0/24. On GigabitEthernet 1/0/2, configure the preauthentication domain as ISP domain **abc**.

```
<Device> system-view
[Device] domain abc
[Device-isp-abc] authorization-attribute acl 3010
[Device-isp-abc] quit
[Device] acl advanced 3010
[Device-acl-ipv4-adv-3010] rule 1 permit ip destination 192.168.0.0 0.0.0.255
[Device-acl-ipv4-adv-3010] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal pre-auth domain abc
[Device–GigabitEthernet1/0/2] quit
```

6. Configure DHCP relay and authorized ARP to allow the host to obtain an IP address from the DHCP:

```
[Device] dhcp enable
[Device] dhcp relay client-information record
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] ip address 20.20.20.1 255.255.255.0
[Device–GigabitEthernet1/0/2] ip address 10.0.0.1 255.255.255.0 sub
[Device-GigabitEthernet1/0/2] dhcp select relay
[Device-GigabitEthernet1/0/2] dhcp relay server-address 192.168.0.112
[Device-GigabitEthernet1/0/2] arp authorized enable
[Device-GigabitEthernet1/0/2] quit
```

7. Configure portal authentication:

   # Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable re-DHCP portal authentication, specify the portal Web server, and configure the BAS-IP as 20.20.20.1.

```
[Device] portal server newpt
[Device-portal-server-newpt] ip 192.168.0.111 key simple portal
[Device-portal-server-newpt] port 50100
[Device-portal-server-newpt] quit
[Device] portal web-server newpt
[Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
[Device-portal-websvr-newpt] quit
[Device] interface gigabitethernet 1/0/2
[Device–GigabitEthernet1/0/2] portal enable method redhcp
[Device–GigabitEthernet1/0/2] portal apply web-server newpt
[Device–GigabitEthernet1/0/2] portal bas-ip 20.20.20.1
[Device–GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify the portal configuration by executing the **display portal interface** command. (Details not shown.)

# Display information about preauthentication portal users.

```
[Device] display portal user pre-auth interface gigabitethernet 1/0/2
Total portal pre-auth users: 1
```

```
MAC                 IP                  VLAN    Interface
0015-e9a6-7cfe      10.10.10.4          --      GigabitEthernet1/0/2
  State: Online
  VPN instance: --
    DHCP IP pool: N/A
    ACL number/name: 3010
    Inbound CAR: N/A
    Outbound CAR: N/A
```

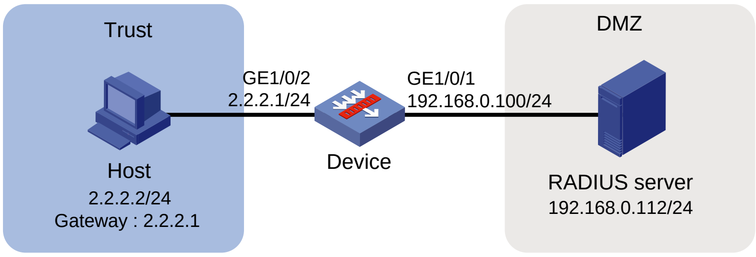
Example: Configuring direct portal authentication using a local portal Web service

Network configuration

As shown in Figure 25, the host is directly connected to the device (the access device). The host is assigned a public IP address either manually or through DHCP. The device acts as both a portal authentication server and a portal Web server. A RADIUS server acts as the authentication and accounting server.

Configure direct portal authentication on the device. Before a user passes portal authentication, the user can access only the portal Web server. After passing portal authentication, the user can access other network resources.

Figure 25 Network diagram



Prerequisites

Customize the authentication pages, compress them to a file, and upload the file to the root directory of the storage medium of the device.

Procedure

1. Configure the RADIUS server correctly to provide authentication and accounting functions. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name trust
   ```

```
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

4. Configure a security policy:

   # Configure a rule named **portallocalout** to allow the device to send packets to the RADIUS server.

   ```
   [Device-security-policy-ip] rule name portallocalout
   [Device-security-policy-ip-2-portallocalout] source-zone local
   [Device-security-policy-ip-2-portallocalout] destination-zone dmz
   [Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
   [Device-security-policy-ip-2-portallocalout] action pass
   [Device-security-policy-ip-2-portallocalout] quit
   ```

   # Configure a rule named **portallocalin1** to allow the device to receive and process the packets from the RADIUS server.

   ```
   [Device-security-policy-ip] rule name portallocalin1
   [Device-security-policy-ip-3-portallocalin1] source-zone dmz
   [Device-security-policy-ip-3-portallocalin1] destination-zone local
   [Device-security-policy-ip-3-portallocalin1] source-ip-host 192.168.0.112
   [Device-security-policy-ip-3-portallocalin1] action pass
   [Device-security-policy-ip-3-portallocalin1] quit
   ```

   # Configure a rule named **portallocalin2** to allow the host to send packets to the portal server.

   ```
   [Device-security-policy-ip] rule name portallocalin2
   [Device-security-policy-ip-4-portallocalin2] source-zone trust
   [Device-security-policy-ip-4-portallocalin2] source-ip-host 2.2.2.2
   [Device-security-policy-ip-4-portallocalin2] destination-zone local
   [Device-security-policy-ip-4-portallocalin2] action pass
   [Device-security-policy-ip-4-portallocalin2] quit
   [Device-security-policy-ip] quit
   ```

5. Configure a RADIUS scheme:

   # Create a RADIUS scheme named **rs1**, and configure the device to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

   ```
   <Device> system-view
   [Device] radius scheme rs1
   [Device-radius-rs1] primary authentication 192.168.0.112
   [Device-radius-rs1] primary accounting 192.168.0.112
   [Device-radius-rs1] key authentication simple radius
   [Device-radius-rs1] key accounting simple radius
   [Device-radius-rs1] user-name-format without-domain
   [Device-radius-rs1] quit
   [Device] radius session-control enable
   ```

6. Configure an authentication domain:

   # Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

   ```
   [Device] domain dm1
   [Device-isp-dm1] authentication portal radius-scheme rs1
   [Device-isp-dm1] authorization portal radius-scheme rs1
   [Device-isp-dm1] accounting portal radius-scheme rs1
   [Device-isp-dm1] quit
   [Device] domain default enable dm1
   ```

7. Configure portal authentication:

   # Configure a portal Web server named **newpt**, and configure the URL of the portal Web server. On GigabitEthernet 1/0/2, enable direct portal authentication, and specify the portal Web server.

   ```
   [Device] portal web-server newpt
   [Device-portal-websvr-newpt] url http://2.2.2.1:2331/portal
   [Device-portal-websvr-newpt] quit
   [Device] interface gigabitethernet 1/0/2
   [Device–GigabitEthernet1/0/2] portal enable method direct
   [Device–GigabitEthernet1/0/2] portal apply web-server newpt
   [Device–GigabitEthernet1/0/2] quit
   ```

   # Create an HTTP-based local portal Web service. Specify file **abc.zip** as the default authentication page file for the local portal Web service (Make sure the file exist under the root directory of the device). Then, set the HTTP listening port number to 2331 for the local portal Web service.

   ```
   [Device] portal local-web-server http
   [Device–portal-local-websvr-http] default-logon-page abc.zip
   [Device–portal-local-websvr-http] tcp-port 2331
   [Device–portal-local-websvr-http] quit
   ```

Verifying the configuration

# Verify that the portal configuration has taken effect.

```
[Device] display portal interface gigabitethernet 1/0/2
 Portal information of GigabitEthernet1/0/2
     NAS-ID profile: Not configured
     Authorization : Strict checking
     ACL           : Disabled
     User profile  : Disabled
     Dual stack    : Disabled
     Dual IP       : Disabled
     Advertisement-push     : Disabled
     Embedded advertisement : Disabled
 IPv4:
     Portal status: Enabled
     Portal authentication type: Direct
     Portal Web server: newpt(active)
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ip: Not configured
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
```

```
     Layer3 source network:
         IP address               Mask

     Destination authenticate subnet:
         IP address               Mask
     Advertisement push: Not configured
 IPv6:
     Portal status: Disabled
     Portal authentication type: Disabled
     Portal Web server: Not configured
     Secondary portal Web server: Not configured
     Portal mac-trigger-server: Not configured
     Authentication domain: Not configured
     Pre-auth domain: Not configured
     Extend-auth domain: Not configured
     User-dhcp-only: Disabled
     Pre-auth IP pool: Not configured
     Max portal users: Not configured
     Bas-ipv6: Not configured
     User detection: Not configured
     Portal temp-pass: Disabled
     Action for server detection:
         Server type    Server name                        Action
         --             --                                 --
     Layer3 source network:
         IP address                                        Prefix length

     Destination authenticate subnet:
         IP address                                        Prefix length
     Advertisement push: Not configured
```

A user can perform portal authentication through a Web page. Before passing the authentication, the user can access only the authentication page **http://2.2.2.1:2331/portal** and all Web requests will be redirected to the authentication page. After passing the authentication, the user can access other network resources.

# After the user passes authentication, display information about the portal user.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: abc
  Portal server: newpt
  State: Online
  VPN instance: --
  MAC               IP                 VLAN   Interface
  0015-e9a6-7cfe    2.2.2.2            --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: N/A
    Inbound CAR: N/A
    Outbound CAR: N/A
```

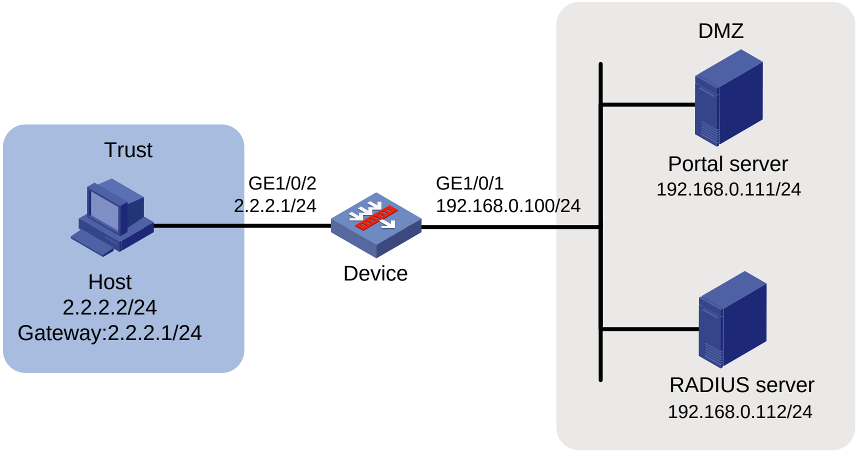
Example: Configuring MAC-based quick portal authentication

Network configuration

As shown in Figure 26, the host accesses the network through a device. The host is assigned a public IP address either manually or through DHCP. An IMC server acts as a portal authentication server, a portal Web server, and a MAC binding server. A RADIUS server acts as the authentication and accounting server. In this example, the IMC server runs IMC 7.1 (E0303) and IMC UAM 7.1 (E0303).

Configure direct portal authentication, so the host can access only the portal Web server before passing the authentication and can access other network resources after passing the authentication.

Figure 26 Network diagram



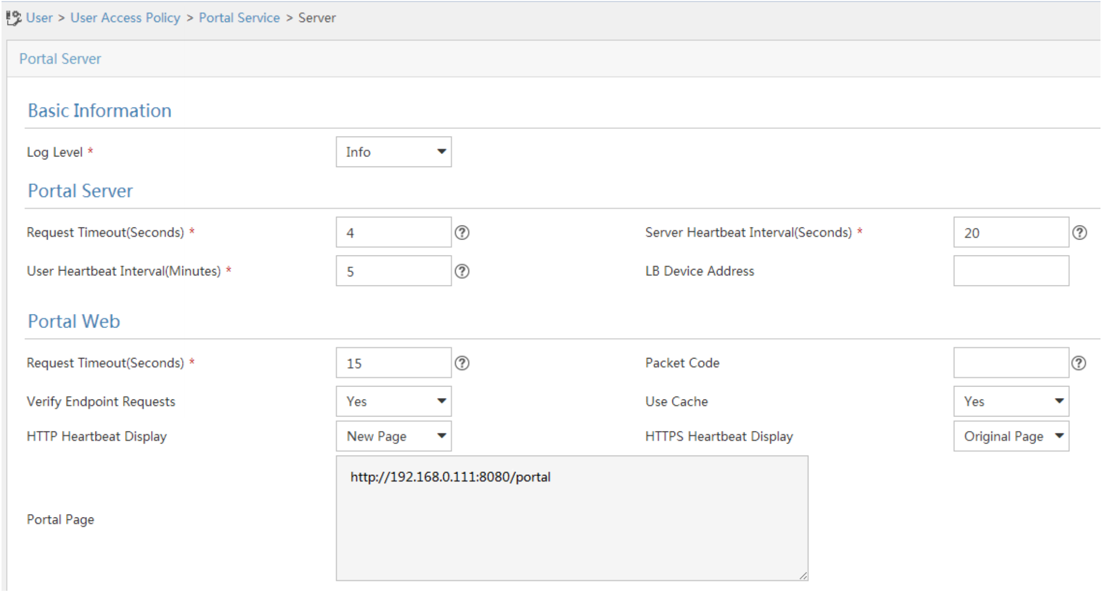
Configuring the RADIUS server

# Configure the RADIUS server correctly to provide authentication and accounting functions. (Details not shown.)

Configuring the portal server

1. Configure the portal authentication server:
   a. Log in to IMC and click the **User** tab.
   b. Select **User Access Policy** > **Portal Service** > **Server** from the navigation tree to open the portal server configuration page, as shown in Figure 27.
   c. Configure the portal server parameters as needed.
      This example uses the default settings.
   d. Click **OK**.

Figure 27 Portal server configuration



User > User Access Policy > Portal Service > Server

Portal Server

Basic Information

Log Level *	Info		

Portal Server

Request Timeout(Seconds) *	4	Server Heartbeat Interval(Seconds) *	20
User Heartbeat Interval(Minutes) *	5	LB Device Address	

Portal Web

Request Timeout(Seconds) *	15	Packet Code	
Verify Endpoint Requests	Yes	Use Cache	Yes
HTTP Heartbeat Display	New Page	HTTPS Heartbeat Display	Original Page
Portal Page	http://192.168.0.111:8080/portal		

2. Configure the IP address group:

   a. Select **User Access Policy** > **Portal Service** > **IP Group** from the navigation tree to open the portal IP address group configuration page.

   b. Click **Add** to open the page as shown in Figure 28.

   c. Enter the IP group name.

   d. Enter the start IP address and end IP address of the IP group.

      Make sure the client IP address (2.2.2.2) is in the IP group.

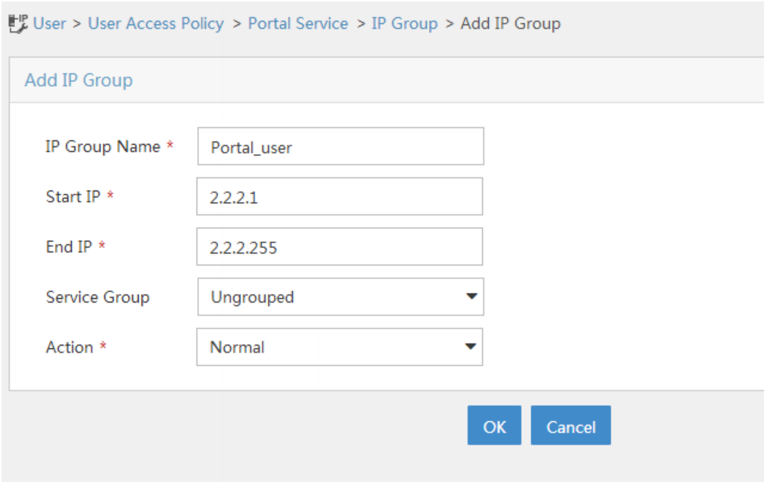
   e. Select a service group.

      This example uses the default group **Ungrouped**.

   f. Select **Normal** from the **Action** list.

   g. Click **OK**.

Figure 28 Adding an IP address group



User > User Access Policy > Portal Service > IP Group > Add IP Group

Add IP Group

IP Group Name *	Portal_user
Start IP *	2.2.2.1
End IP *	2.2.2.255
Service Group	Ungrouped
Action *	Normal

3. Add a portal device:

   a. Select **User Access Policy** > **Portal Service** > **Device** from the navigation tree to open the portal device configuration page.

b. Click **Add** to open the page as shown in Figure 29.

c. Enter the device name.

d. Enter the IP address of the device's interface connected to the host.

e. Set whether to support the portal server heartbeat and user heartbeat functions.

   In this example, select **No** for both **Support Server Heartbeat** and **Support User Heartbeat**.

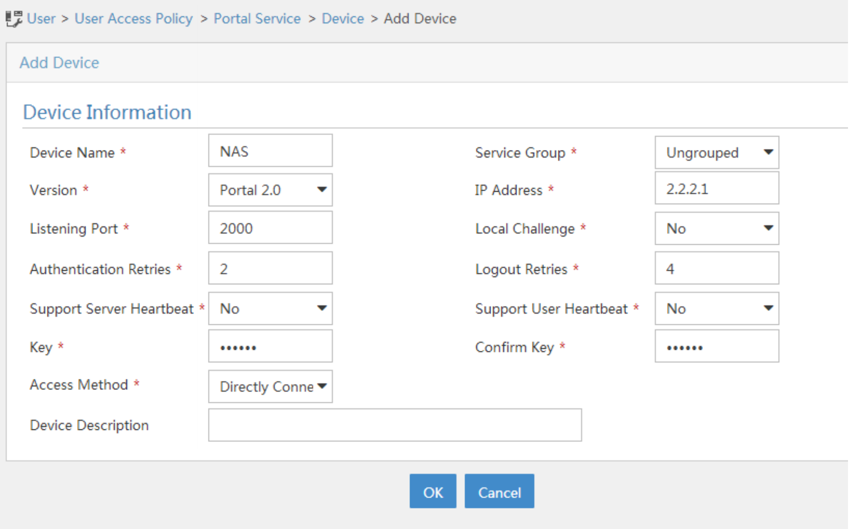
f. Enter the key, which must be the same as that configured on the device.

g. Select **Directly Connected** from the **Access Method** list.

h. Use the default settings for other parameters.

i. Click **OK**.

Figure 29 Adding a portal device


User > User Access Policy > Portal Service > Device > Add Device
OK
Cancel

Add Device

Device Information

Device Name *	NAS	Service Group *	Ungrouped
Version *	Portal 2.0	IP Address *	2.2.2.1
Listening Port *	2000	Local Challenge *	No
Authentication Retries *	2	Logout Retries *	4
Support Server Heartbeat *	No	Support User Heartbeat *	No
Key *	••••••	Confirm Key *	••••••
Access Method *	Directly Conne		
Device Description			

4. Associate the portal device with the IP address group:

a. As shown in Figure 30, click the **Port Group Information Management** icon for device **NAS** to open the port group configuration page.

b. Click **Add** to open the page as shown in Figure 31.

c. Enter the port group name.

d. Select the configured IP address group.

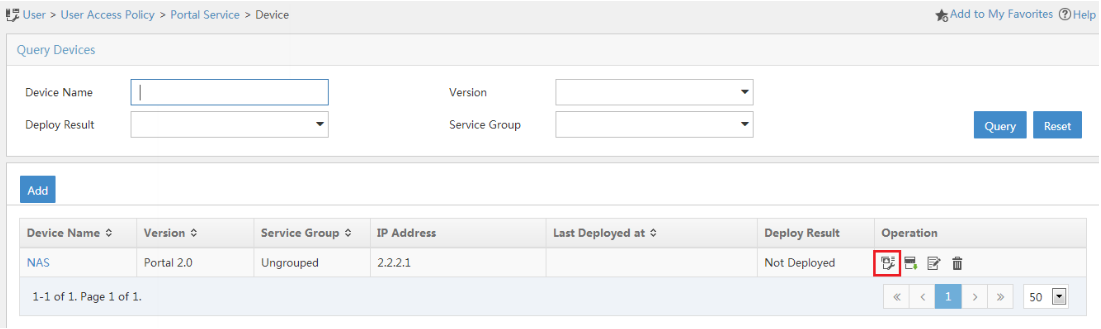
   The IP address used by the user to access the network must be within this IP address group.

e. Select **Supported** for **Transparent Authentication**.

f. Use the default settings for other parameters.

g. Click **OK**.

Figure 30 Device list



User > User Access Policy > Portal Service > Device

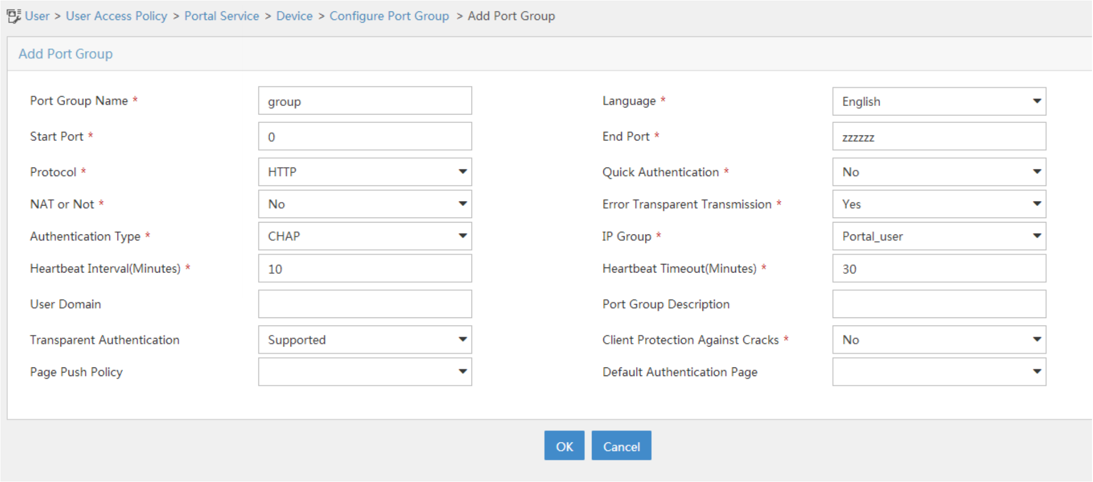
Query Devices

Device Name		Version	
Deploy Result		Service Group	

Device Name	Version	Service Group	IP Address	Last Deployed at	Deploy Result	Operation
NAS	Portal 2.0	Ungrouped	2.2.2.1		Not Deployed	

1-1 of 1. Page 1 of 1.

Figure 31 Adding a port group



User > User Access Policy > Portal Service > Device > Configure Port Group > Add Port Group

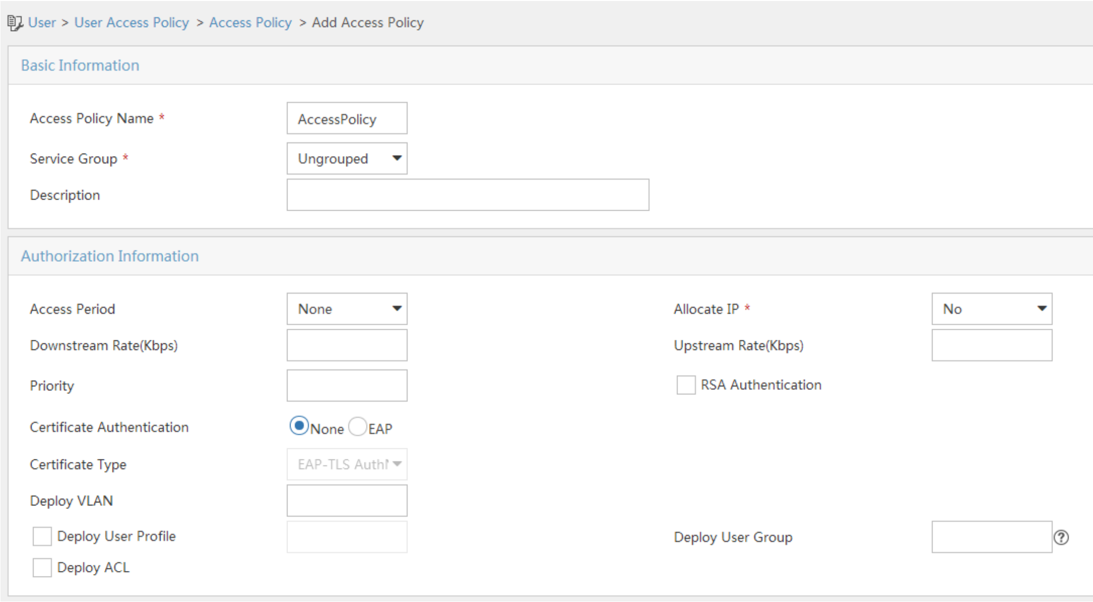
Add Port Group

Port Group Name *	group	Language *	English
Start Port *	0	End Port *	zzzzzz
Protocol *	HTTP	Quick Authentication *	No
NAT or Not *	No	Error Transparent Transmission *	Yes
Authentication Type *	CHAP	IP Group *	Portal_user
Heartbeat Interval(Minutes) *	10	Heartbeat Timeout(Minutes) *	30
User Domain		Port Group Description	
Transparent Authentication	Supported	Client Protection Against Cracks *	No
Page Push Policy		Default Authentication Page	

Configuring the MAC binding server

1. Add an access policy:
   a. Select **User Access Policy** > **Access Policy** from the navigation tree to open the access policy page.
   b. Click **Add** to open the page as shown in Figure 32.
   c. Enter the access policy name.
   d. Select a service group.
   e. Use the default settings for other parameters.
   f. Click **OK**.

Figure 32 Adding an access policy



User > User Access Policy > Access Policy > Add Access Policy

Basic Information

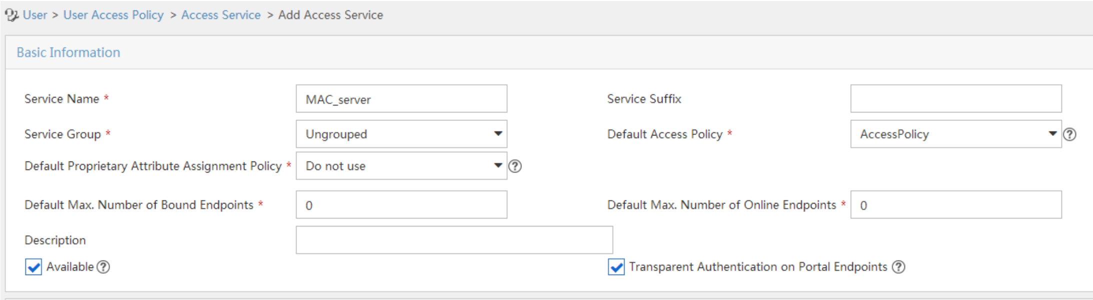
Access Policy Name *	AccessPolicy
Service Group *	Ungrouped
Description	

Authorization Information

Access Period	None	Allocate IP *	No
Downstream Rate(Kbps)		Upstream Rate(Kbps)	
Priority		□ RSA Authentication	
Certificate Authentication	● None ○ EAP		
Certificate Type	EAP-TLS AuthN		
Deploy VLAN			
□ Deploy User Profile		Deploy User Group	
□ Deploy ACL			

2. Add an access service:
   a. Select **User Access Policy** > **Access Service** from the navigation tree to open the access service page.
   b. Click **Add** to open the page as shown in Figure 33.
   c. Enter the service name.
   d. Select the **Transparent Authentication on Portal Endpoints** option.
   e. Use the default settings for other parameters.
   f. Click **OK**.

Figure 33 Adding an access service



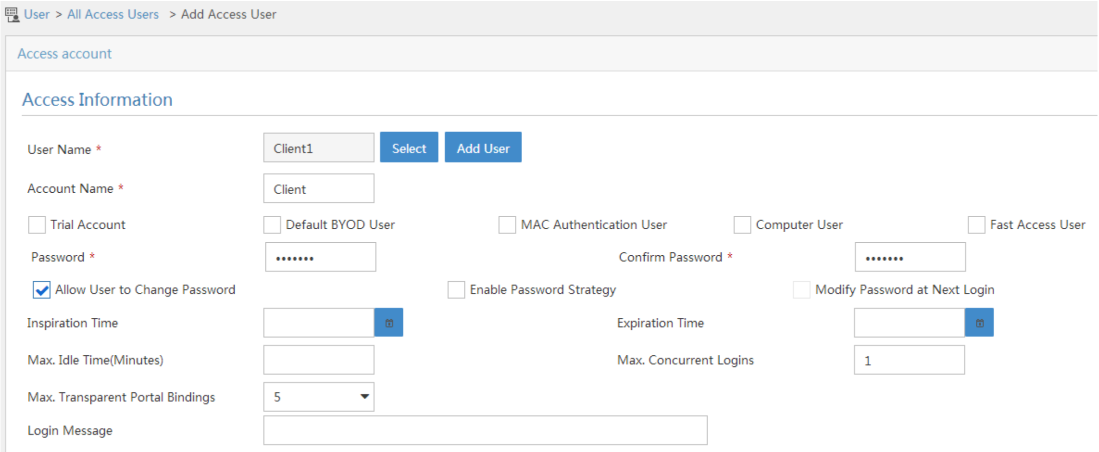
User > User Access Policy > Access Service > Add Access Service

Basic Information

Service Name *	MAC_server	Service Suffix	
Service Group *	Ungrouped	Default Access Policy *	AccessPolicy
Default Proprietary Attribute Assignment Policy *	Do not use		
Default Max. Number of Bound Endpoints *	0	Default Max. Number of Online Endpoints *	0
Description			
☑ Available		☑ Transparent Authentication on Portal Endpoints	

3. Add an access user:
   a. Select **Access User** > **All Access Users** from the navigation tree to open the access user page.
   b. Click **Add** to open the page as shown in Figure 34.
   c. Select an access user.
   d. Set the password.
   e. Select a value from the **Max. Transparent Portal Bindings** list.
   f. Click **OK**.

Figure 34 Adding an access user



User > All Access Users > Add Access User

Access account

Access Information

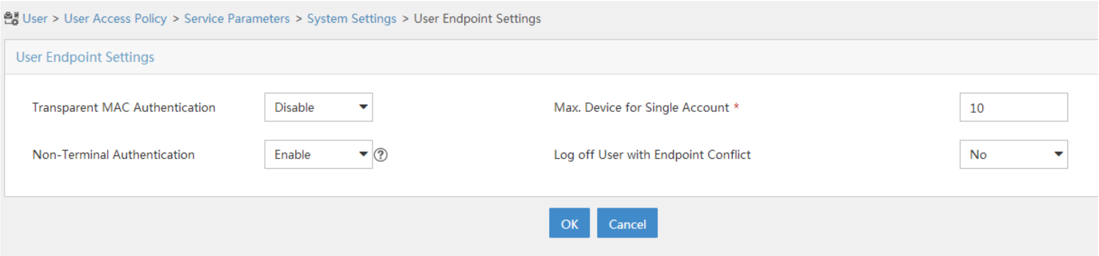
User Name *	Client1		
Account Name *	Client		
Password *	•••••••	Confirm Password *	•••••••
Inspiration Time		Expiration Time	
Max. Idle Time(Minutes)		Max. Concurrent Logins	1
Max. Transparent Portal Bindings	5		
Login Message			

□ Trial Account □ Default BYOD User □ MAC Authentication User □ Computer User □ Fast Access User

☑ Allow User to Change Password □ Enable Password Strategy □ Modify Password at Next Login

4. Configure system parameters:
   a. Select **User Access Policy** > **Service Parameters** > **System Settings** from the navigation tree to open the system settings page.
   b. Click the **Configure** icon ⚙ for **User Endpoint Settings** to open the page as shown in Figure 35.
   c. Select whether to enable transparent portal authentication on non-smart devices.
      In this example, select **Enable** for **Non-Terminal Authentication**.
   d. Click **OK**.
   e. Click the **Configure** icon ⚙ for **Endpoint Aging Time** to open the page as shown in Figure 36.
   f. Set the endpoint aging time as needed.
      This example uses the default value.

Figure 35 Configuring user endpoint settings

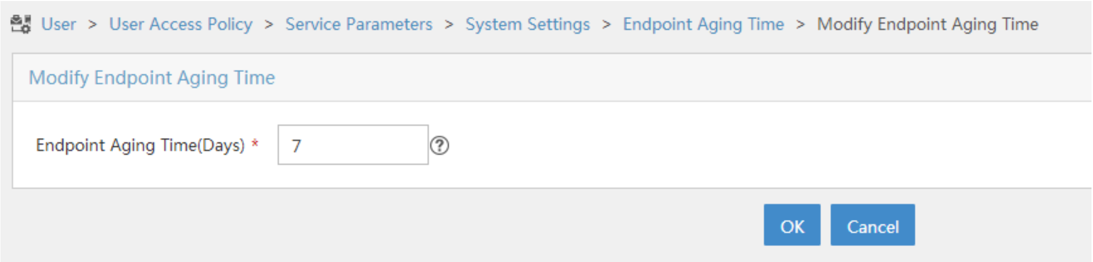


User > User Access Policy > Service Parameters > System Settings > User Endpoint Settings

User Endpoint Settings

Transparent MAC Authentication	Disable	Max. Device for Single Account *	10
Non-Terminal Authentication	Enable	Log off User with Endpoint Conflict	No

Figure 36 Setting the endpoint aging time



User > User Access Policy > Service Parameters > System Settings > Endpoint Aging Time > Modify Endpoint Aging Time

Modify Endpoint Aging Time

Endpoint Aging Time(Days) * 7

Configuring the device

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.100 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
[Device-security-zone-DMZ] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **trust-dmz** to allow the host to access the portal server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule 1 name trust-dmz
[Device-security-policy-ip-1-trust-dmz] source-zone trust
[Device-security-policy-ip-1-trust-dmz] destination-zone dmz
[Device-security-policy-ip-1-trust-dmz] source-ip-host 2.2.2.2
[Device-security-policy-ip-1-trust-dmz] destination-ip-host 192.168.0.111
[Device-security-policy-ip-1-trust-dmz] action pass
[Device-security-policy-ip-1-trust-dmz] quit
```

# Configure a rule named **portallocalout** to allow the device to send packets to the RADIUS server and portal server.

```
[Device-security-policy-ip] rule 2 name portallocalout
[Device-security-policy-ip-2-portallocalout] source-zone local
[Device-security-policy-ip-2-portallocalout] destination-zone dmz
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.111
[Device-security-policy-ip-2-portallocalout] destination-ip-host 192.168.0.112
[Device-security-policy-ip-2-portallocalout] action pass
[Device-security-policy-ip-2-portallocalout] quit
```

# Configure a rule named **portallocalin** to allow the device to receive and process the packets from the RADIUS server, portal server, and MAC-trigger server.

```
[Device-security-policy-ip] rule 3 name portallocalin
[Device-security-policy-ip-3-portallocalin] source-zone dmz
[Device-security-policy-ip-3-portallocalin] destination-zone local
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.111
[Device-security-policy-ip-3-portallocalin] source-ip-host 192.168.0.112
[Device-security-policy-ip-3-portallocalin] action pass
[Device-security-policy-ip-3-portallocalin] quit
[Device-security-policy-ip] quit
```

4. Configure a RADIUS scheme:

# Create a RADIUS scheme named **rs1**, and configure the device to exclude the ISP domain name from the username sent to the RADIUS server, and enable RADIUS session control.

```
<Device> system-view
[Device] radius scheme rs1
[Device-radius-rs1] primary authentication 192.168.0.112
```

```
[Device-radius-rs1] primary accounting 192.168.0.112
[Device-radius-rs1] key authentication simple radius
[Device-radius-rs1] key accounting simple radius
[Device-radius-rs1] user-name-format without-domain
[Device-radius-rs1] quit
[Device] radius session-control enable
```

5. Configure an authentication domain:

   # Create an ISP domain named **dm1**, configure AAA methods for the ISP domain, and specify domain **dm1** as the default ISP domain.

   ```
   [Device] domain dm1
   [Device-isp-dm1] authentication portal radius-scheme rs1
   [Device-isp-dm1] authorization portal radius-scheme rs1
   [Device-isp-dm1] accounting portal radius-scheme rs1
   [Device-isp-dm1] quit
   [Device] domain default enable dm1
   ```

6. Configure portal authentication:

   # Configure a portal authentication server and a portal Web server. On GigabitEthernet 1/0/2, enable direct authentication, specify the portal Web server, and configure the BAS-IP as 2.2.2.1.

   ```
   [Device] portal server newpt
   [Device-portal-server-newpt] ip 192.168.0.111 key simple portal
   [Device-portal-server-newpt] port 50100
   [Device-portal-server-newpt] quit
   [Device] portal web-server newpt
   [Device-portal-websvr-newpt] url http://192.168.0.111:8080/portal
   [Device-portal-websvr-newpt] quit
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] portal enable method direct
   [Device-GigabitEthernet1/0/2] portal apply web-server newpt
   [Device-GigabitEthernet1/0/2] portal bas-ip 2.2.2.1
   [Device-GigabitEthernet1/0/2] quit
   ```

7. Configure MAC-based quick portal authentication:

   # Create the MAC binding server **mts**., set the free-traffic threshold, and specify the IP address of the MAC binding server. Specify the MAC binding server on GigabitEthernet 1/0/2.

   ```
   [Device] portal mac-trigger-server mts
   [Device-portal-mac-trigger-server-mts] free-traffic threshold 1024000
   [Device-portal-mac-trigger-server-mts] ip 192.168.0.111
   [Device-portal-mac-trigger-server-mts] quit
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] portal apply mac-trigger-server mts
   [Device-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Display information about the MAC binding server.

```
[Device] display portal mac-trigger-server name mts
Portal mac-trigger server name: mts
  Version                    : 1.0
  Server type                : IMC
  IP                         : 192.168.0.111
```

```
  Port                       : 50100
  VPN instance               : Not configured
  Aging time                 : 300 seconds
  Free-traffic threshold     : 1024000 bytes
  NAS-Port-Type              : Not configured
  Binding retry times        : 3
  Binding retry interval     : 1 seconds
  Authentication timeout     : 3 minutes
  Local-binding              : Disabled
  Local-binding aging-time   : 12 minutes
  aaa-fail nobinding         : Disabled
  Excluded attribute list    : Not configured
  Cloud-binding              : Disabled
  Cloud-server URL           : Not configured
```

A user can perform portal authentication by using the iNode client or a Web browser. Before passing the authentication, the user can access only the authentication page **http://192.168.0.111:8080/portal**. All Web requests from the user will be redirected to the authentication page. After passing the authentication, the user can access other network resources.

For the first portal authentication, the user is required to enter the username and password. When the user goes offline and then accesses the network again, the user does not need to enter the authentication username and password.

# Display portal user information.

```
[Device] display portal user interface gigabitethernet 1/0/2
Total portal users: 1
Username: Client1
  Portal server: newpt
  State: Online
  VPN instance: N/A
  MAC                IP                 VLAN   Interface
  0015-e9a6-7cfe     2.2.2.2            --     GigabitEthernet1/0/2
  Authorization information:
    DHCP IP pool: N/A
    ACL number/name: N/A
    Inbound CAR: N/A
    Outbound CAR: N/A
```

Troubleshooting portal

No portal authentication page is pushed for users

Symptom

When a user is redirected to the IMC portal authentication server, no portal authentication page or error message is prompted for the user. The login page is blank.

Analysis

The key configured on the portal access device and that configured on the portal authentication server are inconsistent. As a result, packet verification fails, and the portal authentication server refuses to push the authentication page.

Solution

Use the **`display this`** command in portal authentication server view on the access device to check whether a key is configured for the portal authentication server.

- If no key is configured, configure the right key.
- If a key is configured, use the **`ip`** or **`ipv6`** command in the portal authentication server view to correct the key, or correct the key configured for the access device on the portal authentication server.

Cannot log out portal users on the access device

Symptom

You cannot use the **`portal delete-user`** command on the access device to log out a portal user, but the portal user can log out by clicking the **`Disconnect`** button on the portal authentication client.

Analysis

When you execute the **`portal delete-user`** command on the access device to log out a user, the access device sends an unsolicited logout notification message to the portal authentication server. The destination port number in the logout notification is the listening port number of the portal authentication server configured on the access device. If this listening port number is not the actual listening port number configured on the server, the server cannot receive the notification. As a result, the server does not log out the user.

When a user uses the **`Disconnect`** button on the authentication client to log out, the portal authentication server sends an unsolicited logout request message to the access device. The access device uses the source port in the logout request as the destination port in the logout ACK message. As a result, the portal authentication server can definitely receive the logout ACK message and log out the user.

Solution

1. Use the **`display portal server`** command to display the listening port of the portal authentication server configured on the access device.
2. Use the **`portal server`** command in system view to change the listening port number to the actual listening port of the portal authentication server.

Cannot log out portal users on the RADIUS server

Symptom

The access device uses the IMC server as the RADIUS server to perform identity authentication for portal users. You cannot log out the portal users on the RADIUS server.

Analysis

The IMC server uses session control packets to send disconnection requests to the access device. On the access device, the listening UDP port for session control packets is disabled by default. Therefore, the access device cannot receive the portal user logout requests from the RADIUS server.

Solution

On the access device, execute the **`radius session-control enable`** command in system view to enable the RADIUS session control function.

Users logged out by the access device still exist on the portal authentication server

Symptom

After you log out a portal user on the access device, the user still exists on the portal authentication server.

Analysis

When you execute the **`portal delete-user`** command on the access device to log out a user, the access device sends an unsolicited logout notification to the portal authentication server. If the BAS-IP or BAS-IPv6 address carried in the logout notification is different from the portal device IP address specified on the portal authentication server, the portal authentication server discards the logout notification. When sending of the logout notifications times out, the access device logs out the user. However, the portal authentication server does not receive the logout notification successfully, and therefore it regards the user is still online.

Solution

Configure the BAS-IP or BAS-IPv6 attribute on the interface enabled with portal authentication. Make sure the attribute value is the same as the portal device IP address specified on the portal authentication server.

Re-DHCP portal authenticated users cannot log in successfully

Symptom

The device performs re-DHCP portal authentication for users. A user enters the correct username and password, and the client successfully obtains the private and public IP addresses. However, the authentication result for the user is failure.

Analysis

When the access device detects that the client IP address is changed, it sends an unsolicited portal packet to notify of the IP change to the portal authentication server. The portal authentication server notifies of the authentication success only after it receives the IP change notification from both the access device and the client.

If the BAS-IP or BAS-IPv6 address carried in the portal notification packet is different from the portal device IP address specified on the portal authentication server, the portal authentication server discards the portal notification packet. As a result, the portal authentication server considers that the user has failed the authentication.

Solution

Configure the BAS-IP or BAS-IPv6 attribute on the interface enabled with portal authentication. Make sure the attribute value is the same as the portal device IP address specified on the portal authentication server.

Contents

Configuring MAC authentication

About MAC authentication

MAC authentication controls network access by authenticating source MAC addresses on a port. The feature does not require client software, and users do not have to enter a username and password for network access. The device initiates a MAC authentication process when it detects an unknown source MAC address on a MAC authentication-enabled port. If the MAC address passes authentication, the user can access authorized network resources. If the authentication fails, the device marks the MAC address as a silent MAC address, drops the packet, and starts a quiet timer. The device drops all subsequent packets from the MAC address within the quiet time. The quiet mechanism avoids repeated authentication during a short time.

User account policies

MAC authentication supports the following user account policies:

- One MAC-based user account for each user. As shown in Figure 1, the access device uses the source MAC addresses in packets as the usernames and passwords of users for MAC authentication. This policy is suitable for an insecure environment.
- One shared user account for all users. You specify one username and password, which are not necessarily a MAC address, for all MAC authentication users on the access device. This policy is suitable for a secure environment. See Figure 2.

Figure 1 MAC-based user account policy

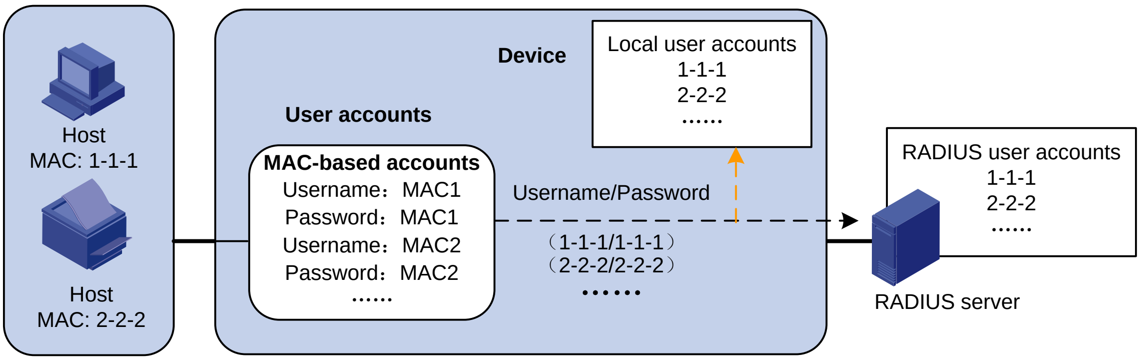
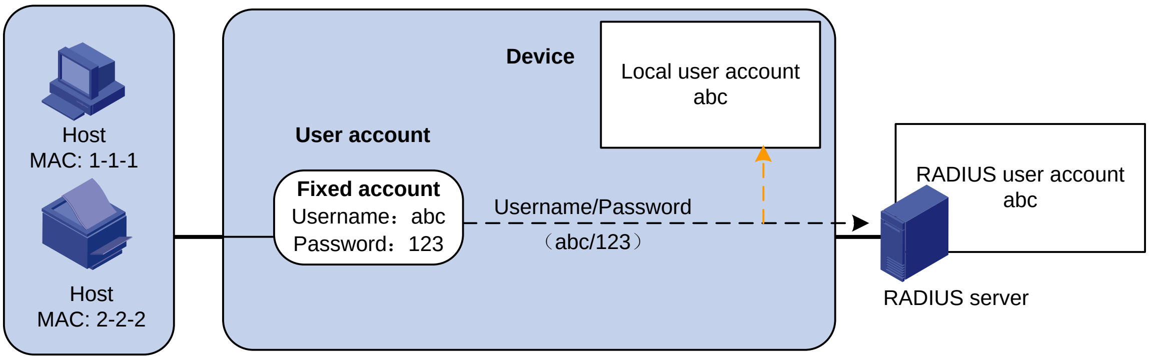


Figure 2 Shared user account policy



Authentication methods

You can perform MAC authentication on the access device (local authentication) or through a RADIUS server.

For more information about configuring local authentication and RADIUS authentication, see "Configuring AAA."

RADIUS authentication

If MAC-based accounts are used, the access device sends the source MAC address of the packet as the username and password to the RADIUS server for authentication.

If a shared account is used, the access device sends the shared account username and password to the RADIUS server for authentication.

Local authentication

If MAC-based accounts are used, the access device uses the source MAC address of the packet as the username and password to search the local account database for a match.

If a shared account is used, the access device uses the shared account username and password to search the local account database for a match.

VLAN assignment

Authorization VLAN

The authorization VLAN controls the access of a MAC authentication user to authorized network resources. The device supports authorization VLANs assigned locally or by a remote server.

IMPORTANT:

Only remote servers can assign tagged authorization VLANs.

Remote VLAN authorization

In remote VLAN authorization, you must configure an authorization VLAN for a user on the remote server. After the user authenticates to the server, the server assigns authorization VLAN information to the device. Then, the device assigns the user access port to the authorization VLAN as a tagged or untagged member.

The device supports assignment of the following authorization VLAN information by the remote server:

- VLAN ID.
- VLAN name, which must be the same as the VLAN description on the access device.
- A string of VLAN IDs and VLAN names.

  In the string, some VLANs are represented by their IDs, and some VLANs are represented by their names.
- VLAN group name.

  For more information about VLAN groups, see VLAN configuration in *Layer 2—LAN Switching Configuration Guide*.
- VLAN ID with a suffix of **t** or **u**.

  The **t** and **u** suffixes require the device to assign the access port to the VLAN as a tagged or untagged member, respectively. For example, **2u** indicates assigning the port to VLAN 2 as an untagged member.

If a VLAN name or VLAN group name is assigned, the device converts the information into a VLAN ID before VLAN assignment.

IMPORTANT:

For a VLAN represented by its VLAN name to be assigned successfully, you must make sure the VLAN has been created on the device.

To assign VLAN IDs with suffixes, make sure the user access port is a hybrid or trunk port.

To ensure a successful assignment, the authorization VLANs assigned by the remote server cannot be any of the following types:

- Dynamically learned VLANs.
- Reserved VLANs.

If the server assigns a group of VLANs, the access device selects a VLAN as described in Table 1.

Table 1 Authorization VLAN selection from a group of VLANs

VLAN information	Authorization VLAN selection
VLANs by IDs VLANs by names VLAN group name	On an access, trunk or hybrid port, the device selects an authorization VLAN from the VLAN group for a user according to the following rules: • If the port does not have online users, the device selects the VLAN with the lowest ID. • If the port has online users, the device examines whether the VLAN that has online users is in the VLAN group. If the VLAN is found in the group, the VLAN is assigned to the user as the authorization VLAN. If the VLAN is not found in the group, the VLAN authorization fails.
VLAN IDs with suffixes	**1.** The device selects the leftmost VLAN ID without a suffix, or the leftmost VLAN ID suffixed by **u** as an untagged VLAN, whichever is more leftmost. **2.** The device assigns the untagged VLAN to the port as the PVID, and it assigns the remaining as tagged VLANs. If no untagged VLAN is assigned, the PVID of the port does not change. The port permits traffic from these tagged and untagged VLANs to pass through. For example, the authentication server sends the string **1u 2t 3** to the access device for a user. The device assigns VLAN 1 as an untagged VLAN and all remaining VLANs (including VLAN 3) as tagged VLANs. VLAN 1 becomes the PVID.

Local VLAN authorization

To perform local VLAN authorization for a user, specify the VLAN ID in the authorization attribute list of the local user account for that user. For each local user, you can specify only one authorization VLAN ID. The user access port is assigned to the VLAN as an untagged member.

IMPORTANT:

Local VLAN authorization does not support assignment of tagged VLANs.

For more information about local user configuration, see "Configuring AAA."

Authorization VLAN manipulation on a MAC authentication-enabled port

Table 2 describes the way the network access device handles authorization VLANs (except for the VLANs specified with suffixes) for MAC authenticated users.

Table 2 VLAN manipulation

Port type	VLAN manipulation
• Access port • Trunk port	• The device assigns the port to the first authenticated user's authorization VLAN and sets the VLAN as the PVID if that authorization VLAN has the untagged attribute.

Port type	VLAN manipulation
• Hybrid port	• If the authorization VLAN has the tagged attribute, the device assigns the port to the authorization VLAN without changing its PVID. **NOTE:** The tagged attribute is supported only on trunk and hybrid ports.

IMPORTANT:

- If the users are attached to a port whose link type is access, make sure the authorization VLAN assigned by the server has the untagged attribute. VLAN assignment will fail if the server issues a VLAN that has the tagged attribute.
- As a best practice to enhance network security, do not use the **`port hybrid vlan`** command to assign a hybrid port to an authorization VLAN as a tagged member.

For a MAC authenticated user to access the network on a hybrid port when no authorization VLAN is configured for the user, perform one of the following tasks:

- If the port receives tagged authentication packets from the user in a VLAN, use the **`port hybrid vlan`** command to configure the port as a tagged member in the VLAN.
- If the port receives untagged authentication packets from the user in a VLAN, use the **`port hybrid vlan`** command to configure the port as an untagged member in the VLAN.

Guest VLAN

The MAC authentication guest VLAN on a port accommodates users that have failed MAC authentication for any reason other than server unreachable. For example, the VLAN accommodates users for which invalid passwords are entered.

You can deploy a limited set of network resources in the MAC authentication guest VLAN. For example, a software server for downloading software and system patches.

A hybrid port is always assigned to a MAC authentication guest VLAN as an untagged member. After the assignment, do not reconfigure the port as a tagged member in the VLAN.

The device reauthenticates users in the MAC authentication guest VLAN at a specific interval. Table 3 shows the way that the network access device handles guest VLANs for MAC authentication users.

Table 3 VLAN manipulation

Authentication status	VLAN manipulation
A user in the MAC authentication guest VLAN fails MAC authentication.	The user is still in the MAC authentication guest VLAN.
A user in the MAC authentication guest VLAN passes MAC authentication.	The device remaps the MAC address of the user to the authorization VLAN assigned by the authentication server. If no authorization VLAN is configured for the user on the authentication server, the device remaps the MAC address of the user to the PVID of the port.

Critical VLAN

The MAC authentication critical VLAN on a port accommodates users that have failed MAC authentication because no RADIUS authentication servers are reachable. Users in a MAC authentication critical VLAN can access only network resources in the critical VLAN.

The critical VLAN feature takes effect when MAC authentication is performed only through RADIUS servers. If a MAC authentication user fails local authentication after RADIUS authentication, the user is not assigned to the critical VLAN. For more information about the authentication methods, see "Configuring AAA."

Table 4 shows the way that the network access device handles critical VLANs for MAC authentication users.

Table 4 VLAN manipulation

Authentication status	VLAN manipulation
A user fails MAC authentication because all the RADIUS servers are unreachable.	The device maps the MAC address of the user to the MAC authentication critical VLAN. The user is still in the MAC authentication critical VLAN if the user fails MAC reauthentication because all the RADIUS servers are unreachable. If no MAC authentication critical VLAN is configured, the device maps the MAC address of the user to the PVID of the port.
A user in the MAC authentication critical VLAN fails MAC authentication for any reason other than server unreachable.	If a guest VLAN has been configured, the device maps the MAC address of the user to the guest VLAN. If no guest VLAN is configured, the device maps the MAC address of the user to the PVID of the port.
A user in the MAC authentication critical VLAN passes MAC authentication.	The device remaps the MAC address of the user to the authorization VLAN assigned by the authentication server. If no authorization VLAN is configured for the user on the authentication server, the device remaps the MAC address of the user to the PVID of the access port.

User profile assignment

You can specify a user profile in the user account for a MAC authentication user on the authentication server to control the user's access to network resources. After the user passes MAC authentication, the authentication server assigns the user profile to the user to filter traffic for this user.

The authentication server can be the local access device or a RADIUS server. In either case, the server only specifies the user profile name. You must configure the user profile on the access device.

To change the user's access permissions, you can use one of the following methods:

- Modify the user profile configuration on the access device.
- Specify another user profile for the user on the authentication server.

For more information about user profiles, see "Configuring user profiles."

Restrictions and guidelines: MAC authentication configuration

Do not change the link type of a port when the MAC authentication guest VLAN or critical VLAN on the port has users.

If the MAC address that has failed authentication is a static MAC address or a MAC address that has passed any security authentication, the device does not mark the MAC address as a silent address.

MAC authentication tasks at a glance

To configure MAC authentication, perform the following tasks:

1. Enabling MAC authentication
2. Configure basic MAC authentication features
   - Specifying a MAC authentication domain
   - Configuring the user account format
   - (Optional.) Configuring MAC authentication timers
3. (Optional.) Configuring MAC authentication VLAN assignment
   - Configuring a MAC authentication guest VLAN
   - Configuring a MAC authentication critical VLAN
4. (Optional.) Configuring other MAC authentication features
   - Setting the maximum number of concurrent MAC authentication users on a port
   - Enabling MAC authentication multi-VLAN mode on a port

     Perform this task to not reauthenticate online users when VLAN changes occur on a port.
   - Configuring the keep-online feature
   - Logging off MAC authentication users
   - Enabling MAC authentication user logging

Prerequisites for MAC authentication

Before you configure MAC authentication, complete the following tasks:

1. Configure an ISP domain and specify an AAA method. For more information, see "Configuring AAA."
   - For local authentication, you must also create local user accounts (including usernames and passwords) and specify the **lan-access** service for local users.
   - For RADIUS authentication, make sure the device and the RADIUS server can reach each other and create user accounts on the RADIUS server. If you are using MAC-based accounts, make sure the username and password for each account are the same as the MAC address of each MAC authentication user.

Enabling MAC authentication

Restrictions and guidelines

For MAC authentication to take effect on a port, you must enable this feature globally and on the port.

You cannot enable MAC authentication on a port that is in a link aggregation group.

Procedure

1. Enter system view.

   `system-view`
2. Enable MAC authentication globally.

   `mac-authentication`

   By default, MAC authentication is disabled globally.
3. Enter Layer 2 Ethernet interface view.

   `interface` *`interface-type interface-number`*

4. Enable MAC authentication on the port.

   mac-authentication

   By default, MAC authentication is disabled on a port.

Specifying a MAC authentication domain

About this task

By default, MAC authentication users are in the system default authentication domain. To implement different access policies for users, you can use one of the following methods to specify authentication domains for MAC authentication users:

- Specify a global authentication domain in system view. This domain setting applies to all ports enabled with MAC authentication.
- Specify an authentication domain for an individual port in Layer 2 Ethernet interface view.

MAC authentication chooses an authentication domain for users on a port in this order: the port-specific domain, the global domain, and the default domain. For more information about authentication domains, see "Configuring AAA."

Procedure

1. Enter system view.

   system-view

2. Specify an authentication domain for MAC authentication users.

   - In system view:

     mac-authentication domain *domain-name*

   - In Layer 2 Ethernet interface view:

     interface *interface-type interface-number*

     mac-authentication domain *domain-name*

   By default, the system default authentication domain is used for MAC authentication users.

Configuring the user account format

1. Enter system view.

   system-view

2. Configure the MAC authentication user account format.

   - Use one MAC-based user account for each user.

     mac-authentication user-name-format mac-address [{ **with-hyphen** [**six-section** | **three-section**] | **without-hyphen** } [**lowercase** | **uppercase**]]

   - Use one shared user account for all users.

     mac-authentication user-name-format fixed [**account** *name*] [**password** { **cipher** | **simple** } *string*]

   By default, the device uses the MAC address of a user as the username and password for MAC authentication. The MAC address is in hexadecimal notation without hyphens, and letters are in lower case.

Configuring MAC authentication timers

About this task

MAC authentication uses the following timers:

- **Offline detect timer**—Sets the interval that the device waits for traffic from a user before the device determines that the user is idle. If the device has not received traffic from a user before the timer expires, the device logs off that user and requests the accounting server to stop accounting for the user.
- **Quiet timer**—Sets the interval that the device must wait before the device can perform MAC authentication for a user that has failed MAC authentication. All packets from the MAC address are dropped during the quiet time. This quiet mechanism prevents repeated authentication from affecting system performance.
- **Server timeout timer**—Sets the interval that the device waits for a response from a RADIUS server before the device determines that the RADIUS server is unavailable. If the timer expires during MAC authentication, the user cannot access the network.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure MAC authentication timers.

   ```
   mac-authentication timer { offline-detect offline-detect-value |
   quiet quiet-value | server-timeout server-timeout-value }
   ```

   By default, the offline detect timer is 300 seconds, the quiet timer is 60 seconds, and the server timeout timer is 100 seconds.

Configuring a MAC authentication guest VLAN

Restrictions and guidelines

When you configure the MAC authentication guest VLAN on a port, follow the guidelines in Table 5.

Table 5 Relationships of the MAC authentication guest VLAN with other security features

Feature	Relationship description	Reference
Quiet feature of MAC authentication	The MAC authentication guest VLAN feature has higher priority. When a user fails MAC authentication, the user can access the resources in the guest VLAN. The user's MAC address is not marked as a silent MAC address.	See "Configuring MAC authentication timers."

Prerequisites

Before you configure the MAC authentication guest VLAN on a port, complete the following tasks:

- Create the VLAN to be specified as the MAC authentication guest VLAN.
- Configure the port as a hybrid port, and configure the VLAN as an untagged member on the port.

For information about VLAN configuration, see *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter Layer 2 Ethernet interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify the MAC authentication guest VLAN on the port.

   ```
   mac-authentication guest-vlan guest-vlan-id
   ```

   By default, no MAC authentication guest VLAN is specified on a port.

   You can configure only one MAC authentication guest VLAN on a port. The MAC authentication guest VLANs on different ports can be different.

4. Set the authentication interval for users in the MAC authentication guest VLAN.

   ```
   mac-authentication guest-vlan auth-period period-value
   ```

   The default setting is 30 seconds.

Configuring a MAC authentication critical VLAN

Restrictions and guidelines

When you configure the MAC authentication critical VLAN on a port, follow the guidelines in Table 6.

Table 6 Relationships of the MAC authentication critical VLAN with other security features

Feature	Relationship description	Reference
Quiet feature of MAC authentication	The MAC authentication critical VLAN feature has higher priority. When a user fails MAC authentication because no RADIUS authentication server is reachable, the user can access the resources in the critical VLAN. The user's MAC address is not marked as a silent MAC address.	See "Configuring MAC authentication timers."

Prerequisites

Before you configure the MAC authentication critical VLAN on a port, complete the following tasks:

- Create the VLAN to be specified as the MAC authentication critical VLAN.
- Configure the port as a hybrid port, and configure the VLAN as an untagged member on the port.

For information about VLAN configuration, see *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter Layer 2 Ethernet interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify the MAC authentication critical VLAN on the port.

   ```
   mac-authentication critical vlan critical-vlan-id
   ```

   By default, no MAC authentication critical VLAN is specified on a port.

   You can configure only one MAC authentication critical VLAN on a port. The MAC authentication critical VLANs on different ports can be different.

Setting the maximum number of concurrent MAC authentication users on a port

About this task

Perform this task to prevent the system resources from being overused.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 2 Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Set the maximum number of concurrent MAC authentication users on the port.

   `mac-authentication max-user` *`max-number`*

   The default setting is 4294967295.

Enabling MAC authentication multi-VLAN mode on a port

About this task

The MAC authentication multi-VLAN mode prevents an authenticated online user from service interruption caused by VLAN changes on a port. When the port receives a packet sourced from the user in a VLAN not matching the existing MAC-VLAN mapping, the device neither logs off the user nor reauthenticates the user. The device creates a new MAC-VLAN mapping for the user, and traffic transmission is not interrupted. The original MAC-VLAN mapping for the user remains on the device until it dynamically ages out. As a best practice, configure this feature on hybrid or trunk ports.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 2 Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Enable MAC authentication multi-VLAN mode.

   `mac-authentication host-mode multi-vlan`

   By default, this feature is disabled on a port. When the port receives a packet sourced from an authenticated user in a VLAN not matching the existing MAC-VLAN mapping, the device logs off and reauthenticates the user.

Configuring the keep-online feature

About this task

Periodic MAC reauthentication tracks the connection status of online users, and updates the authorization attributes assigned by the RADIUS server. The attributes include VLAN.

The device reauthenticates an online MAC authentication user periodically only after it receives the termination action **Radius-request** from the authentication server for this user. The Session-Timeout attribute (session timeout period) assigned by the server is the reauthentication interval. To display the server-assigned Session-Timeout and Termination-Action attributes, use the

display mac-authentication connection command. Support for the server configuration and assignment of Session-Timeout and Termination-Action attributes depends on the server model.

The keep-online feature enables the device to keep the MAC authentication users online when no server is reachable for MAC reauthentication.

Restrictions and guidelines

In a fast-recovery network, you can use the keep-online feature to prevent MAC authentication users from coming online and going offline frequently.

Any modification to the MAC authentication domain or user account format setting does not affect the reauthentication of online MAC authentication users. The modified setting takes effect only on MAC authentication users that come online after the modification.

Procedure

1. Enter system view.

   system-view
2. Enter Layer 2 Ethernet interface view.

   interface *interface-type interface-number*
3. Enable the keep-online feature for authenticated MAC authentication users on the port.

   mac-authentication re-authenticate server-unreachable keep-online

   By default, the keep-online feature is disabled. The device logs off online MAC authentication users if no server is reachable for MAC reauthentication.

Logging off MAC authentication users

About this task

Perform this task to log off specified MAC authentication users and clear information about these users from the device. These users must perform MAC authentication to come online again.

Restrictions and guidelines

With an interface specified, the **reset mac-authentication access-user** command logs off all MAC authentication users on the specified interface.

With a MAC address specified, the **reset mac-authentication access-user** command logs off the MAC authentication user with the specified MAC address.

With a VLAN specified, the **reset mac-authentication access-user** command logs off all MAC authentication users in the specified VLAN, including:

- Users that are performing MAC authentication in the specified VLAN.
- Users that have passed MAC authentication and have been assigned the specified VLAN as their authorization VLAN by the server.
- Users that stay in the specified VLAN after they have passed MAC authentication, because they have not been assigned an authorization VLAN yet.

To identify the VLAN in which a user is staying, use the **display mac-address** command.

With a VSI specified, the **reset mac-authentication access-user** command logs off all MAC authentication users in the specified VSI.

Procedure

To log off MAC authentication users, execute the following command in user view:

reset mac-authentication access-user [**interface** *interface-type interface-number* | **mac** *mac-address* | **username** *username* | **vlan** *vlan-id* | **vsi** *vsi-name*]

Enabling MAC authentication user logging

About this task

This feature enables the device to generate logs about MAC authentication users and send the logs to the information center. For the logs to be output correctly, you must also configure the information center on the device. For more information about information center configuration, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

To prevent excessive MAC authentication user log entries, use this feature only if you need to analyze abnormal MAC authentication user logins or logouts.

Procedure

1. Enter system view.

   system-view

2. Enable MAC authentication user logging.

   mac-authentication access-user log enable [**failed-login** | **logoff** | **successful-login**] *

   By default, MAC authentication user logging is disabled.

   If you do not specify any parameters, this command enables all types of MAC authentication user logs.

Display and maintenance commands for MAC authentication

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display MAC authentication information.	**display mac-authentication** [**interface** *interface-type interface-number*]

Display MAC authentication connections.	**display mac-authentication connection** [**interface** *interface-type interface-number* | **slot** *slot-number* | **user-mac** *mac-address* | **user-name** *user-name*]
Clear MAC authentication statistics.	**reset mac-authentication statistics** [**interface** *interface-type interface-number*]
Remove users from the MAC authentication critical VLAN on a port.	**reset mac-authentication critical-vlan interface** *interface-type interface-number* [**mac-address** *mac-address*]
Remove users from the MAC authentication guest VLAN on a port.	**reset mac-authentication guest-vlan interface** *interface-type interface-number* [**mac-address** *mac-address*]

MAC authentication configuration examples

Example: Configuring local MAC authentication

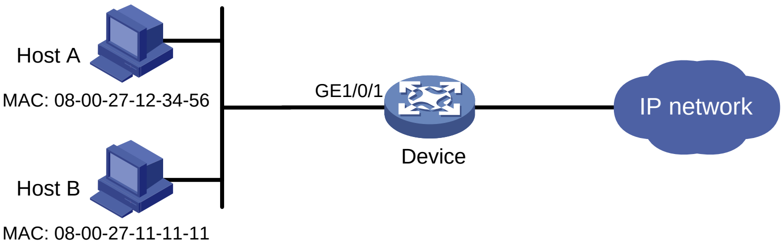
Network configuration

As shown in Figure 3, the device performs local MAC authentication on GigabitEthernet 1/0/1 to control Internet access of users.

Configure the device to meet the following requirements:

- Detect whether a user has gone offline every 180 seconds.
- Deny a user for 180 seconds if the user fails MAC authentication.
- Authenticate all users in ISP domain **bbb**.
- Use the MAC address of each user as the username and password for authentication. The MAC addresses are in hexadecimal notation with hyphens, and letters are in lower case.

Figure 3 Network diagram



Procedure

# Add a network access local user. In this example, configure both the username and password as Host A's MAC address 08-00-27-12-34-56.

```
<Device> system-view
[Device] local-user 08-00-27-12-34-56 class network
[Device-luser-network-08-00-27-12-34-56] password simple 08-00-27-12-34-56
```

# Specify the LAN access service for the user.

```
[Device-luser-network-08-00-27-12-34-56] service-type lan-access
[Device-luser-network-08-00-27-12-34-56] quit
```

# Configure ISP domain **bbb** to perform local authentication for LAN users.

```
[Device] domain bbb
[Device-isp-bbb] authentication lan-access local
[Device-isp-bbb] quit
```

# Enable MAC authentication on GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] mac-authentication
[Device-GigabitEthernet1/0/1] quit
```

# Specify ISP domain **bbb** as the MAC authentication domain.

```
[Device] mac-authentication domain bbb
```

# Configure MAC authentication timers.

```
[Device] mac-authentication timer offline-detect 180
[Device] mac-authentication timer quiet 180
```

# Configure MAC authentication to use MAC-based accounts. Each MAC address is in hexadecimal notation with hyphens, and letters are in lower case.

```
[Device] mac-authentication user-name-format mac-address with-hyphen lowercase
```

# Enable MAC authentication globally.

```
[Device] mac-authentication
```

Verifying the configuration

# Display MAC authentication settings and statistics to verify your configuration.

```
<Device> display mac-authentication
Global MAC authentication parameters:
   MAC authentication     : Enabled
   User name format       : MAC address in lowercase(xx-xx-xx-xx-xx-xx)
           Username       : mac
           Password       : Not configured
   Offline detect period  : 180 s
   Quiet period           : 180 s
```

```
  Server timeout         : 100 s
  Authentication domain  : bbb
Online MAC-auth wired users    : 1

Silent MAC users:
         MAC address       VLAN ID  From port               Port index
         0800-2711-1111    8        Gigabitethernet1/0/1    1

Gigabitethernet1/0/1 is link-up
  MAC authentication         : Enabled
  Carry User-IP              : Disabled
  Authentication domain      : Not configured
  Auth-delay timer           : Disabled
  Re-auth server-unreachable : Logoff
  Guest VLAN                 : Not configured
  Guest VLAN auth-period     : 30 s
  Critical VLAN              : Not configured
  Critical voice VLAN        : Disabled
  Host mode                  : Single VLAN
  Max online users           : 4294967295
  Authentication attempts    : successful 1, failed 0
  Current online users       : 1
         MAC address       Auth state
         0800-2712-3456    Authenticated
```

The output shows that Host A has passed MAC authentication and has come online. Host B failed MAC authentication and its MAC address is marked as a silent MAC address.

Example: Configuring RADIUS-based MAC authentication

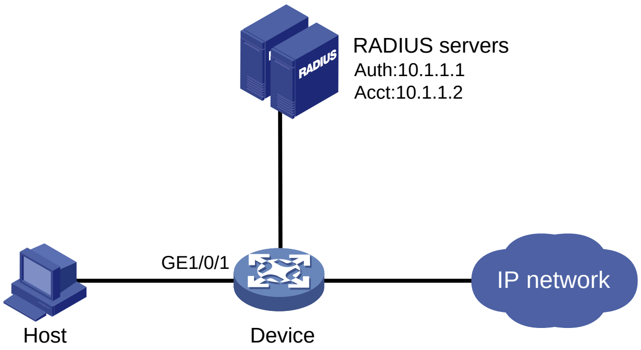
Network configuration

As shown in Figure 4, the device uses RADIUS servers to perform authentication, authorization, and accounting for users.

To control user access to the Internet by MAC authentication, perform the following tasks:

- Enable MAC authentication globally and on GigabitEthernet 1/0/1.
- Configure the device to detect whether a user has gone offline every 180 seconds.
- Configure the device to deny a user for 180 seconds if the user fails MAC authentication.
- Configure all users to belong to ISP domain **bbb**.
- Use a shared user account for all users, with username **aaa** and password **123456TESTplat&!TESTplat&!**.

Figure 4 Network diagram



Procedure

Make sure the RADIUS server and the access device can reach each other.

1. Configure the RADIUS servers to provide authentication, authorization, and accounting services. Create a shared account with username **aaa** and password **123456TESTplat&!TESTplat&!** for MAC authentication users. (Details not shown.)
2. Configure RADIUS-based MAC authentication on the device:

   # Configure a RADIUS scheme.

   ```
   <Device> system-view
   [Device] radius scheme 2000
   [Device-radius-2000] primary authentication 10.1.1.1 1812
   [Device-radius-2000] primary accounting 10.1.1.2 1813
   [Device-radius-2000] key authentication simple abc
   [Device-radius-2000] key accounting simple abc
   [Device-radius-2000] user-name-format without-domain
   [Device-radius-2000] quit
   ```

   # Apply the RADIUS scheme to ISP domain **bbb** for authentication, authorization, and accounting.

   ```
   [Device] domain bbb
   [Device-isp-bbb] authentication default radius-scheme 2000
   [Device-isp-bbb] authorization default radius-scheme 2000
   [Device-isp-bbb] accounting default radius-scheme 2000
   [Device-isp-bbb] quit
   ```

   # Enable MAC authentication on GigabitEthernet 1/0/1.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] mac-authentication
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Specify the MAC authentication domain as ISP domain **bbb**.

   ```
   [Device] mac-authentication domain bbb
   ```

   # Set MAC authentication timers.

   ```
   [Device] mac-authentication timer offline-detect 180
   [Device] mac-authentication timer quiet 180
   ```

   # Specify username **aaa** and password **123456TESTplat&!TESTplat&!** in plain text for the account shared by MAC authentication users.

   ```
   [Device] mac-authentication user-name-format fixed account aaa password simple
   123456TESTplat&!TESTplat&!
   ```

   # Enable MAC authentication globally.

```
[Device] mac-authentication
```

Verifying the configuration

# Verify the MAC authentication configuration.

```
<Device> display mac-authentication
Global MAC authentication parameters:
   MAC authentication     : Enabled
   Username format        : Fixed account
           Username       : aaa
           Password       : ******
   Offline detect period  : 180 s
   Quiet period           : 180 s
   Server timeout         : 100 s
   Authentication domain  : bbb
 Online MAC-auth wired users    : 1

 Silent MAC users:
          MAC address       VLAN ID  From port               Port index

 GigabitEthernet1/0/1  is link-up
   MAC authentication         : Enabled
   Carry User-IP              : Disabled
   Authentication domain      : Not configured
   Auth-delay timer           : Disabled
   Re-auth server-unreachable : Logoff
   Guest VLAN                 : Not configured
   Guest VLAN auth-period     : 30 s
   Critical VLAN              : Not configured
   Critical voice VLAN        : Disabled
   Host mode                  : Single VLAN
   Max online users           : 4294967295
   Authentication attempts    : successful 1, failed 0
   Current online users       : 1
          MAC address       Auth state
          0800-2712-3456    Authenticated
```

Contents

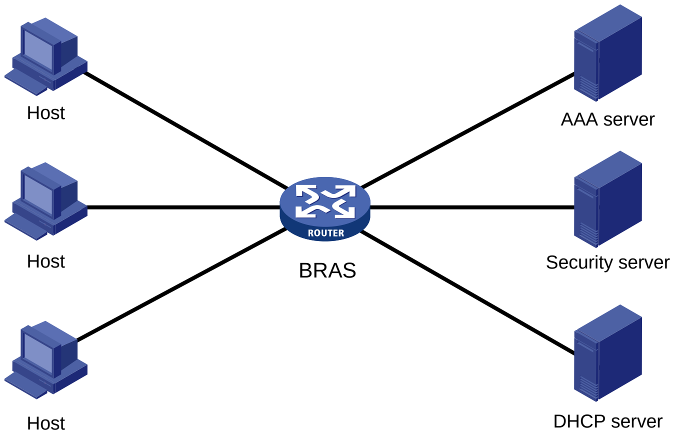
Configuring IPoE

About IPoE

IP over Ethernet (IPoE) enables a BRAS to connect and authenticate users over IPoE connections.

As shown in Figure 1, a BRAS connects hosts over IPoE connections, and provides AAA, security, and DHCP services for the hosts. This solution does not require the hosts to install any client software.

Figure 1 IPoE network diagram



IPoE access modes

IPoE supports Layer 2 and Layer 3 access modes.

- Layer 2 access mode

  Hosts directly access the BRAS. The hosts connect to the BRAS directly or through Layer 2 devices. The BRAS uses MAC addresses to identify the hosts.

- Layer 3 access mode

  Hosts use routing to access the BRAS. The hosts connect to the BRAS directly or through Layer 3 devices. When a Layer 3 device resides between the hosts and the BRAS, the source MAC address of packets received by the BRAS is the MAC address of the Layer 3 device. Therefore, the BRAS uses IP addresses or VLAN IDs to identify hosts.

IPoE users

IPoE users include individual users and leased users.

Individual users

Individual users use independent IPoE services. The BRAS authenticates, authorizes, and bills individual users based on user location and packet information. Individual users include dynamic and static individual users.

- Dynamic individual users

  IPoE defines the following dynamic individual users:

  - **DHCP user**—Sends DHCP packets to trigger IPoE session establishment.
  - **IPv6-ND-RS user**—Sends IPv6 ND RS packets to trigger IPoE session establishment.

  - **Unclassified-IP user**—Sends packets other than DHCP and IPv6 ND RS packets to trigger IPoE session establishment.
- Static individual users

  Static individual users trigger IPoE session establishment by sending IP packets. If an IP packet matches a manually configured IPoE session, the BRAS authenticates the user and establishes an IPoE session.

Leased users

Leased users include the following types:

- **Interface-leased user**—Represents hosts that rent the same interface.
- **Subnet-leased user**—Represents hosts that rent a subnet of an interface.

The BRAS automatically uses the credentials configured for a leased user to perform authentication. Users are not required to send IP packets to trigger authentication.

IPoE session

IPoE sessions include dynamic and static sessions.

Dynamic IPoE session

IPoE sessions established for dynamic individual users are dynamic IPoE sessions.

The BRAS disconnects a dynamic IPoE session in one of the following cases:

- The AAA-authorized service expires.
- The AAA server logs out the user.
- The user traffic is less than the authorized traffic during the idle-timeout time.
- The BRAS cannot detect the user after the number of detection attempts reaches the maximum.
- The IP address lease expires.
- The IPoE session is restarted.
- The access interface goes down.

Static IPoE session

IPoE sessions established for static individual users, interface-leased users, and subnet-leased users are static IPoE sessions.

The BRAS creates a static IPoE session based on configured information after you enable IPoE on an interface. The BRAS initiates user authentication based on the configured username and password upon receiving IP packets from static individual users. Static IPoE sessions can only be deleted manually at the CLI.

For static individual users, the BRAS creates a static IPoE session based on configured information after you enable IPoE on an interface. The BRAS initiates user authentication based on the configured username and password upon receiving IP packets from static individual users.

For interface-leased users and subnet-leased users, the BRAS creates a static IPoE session based on configured information after you enable IPoE on an interface. The BRAS initiates user authentication based on the configured username and password.

IPoE addressing

IPoE addressing varies with user types.

- DHCP users obtain IP addresses in the following sequence:
  - Obtain IP addresses from the AAA-authorized IP address pool.

    - Obtain IP addresses from the IP address pool configured in the ISP domain if the AAA server does not authorize any IP address pools.
    - Obtain IP addresses in the same network segment as the interface IP address if no IP address pool is configured in the ISP domain.
- IPv6-ND-RS users obtain the AAA-authorized IPv6 prefix from IPoE, and generate an IPv6 address based on the prefix. If no AAA-authorized IPv6 prefix exists, the user adopts the first 64-bit IPv6 prefix of the interface to generate an IPv6 address.
- Other users adopt static IP addresses or obtain IP addresses from the DHCP server without using IPoE.

IPoE access procedure

IPoE access includes the following steps:

1. The BRAS initiates authentication.

   The BRAS obtains information from user packets or IPoE sessions statically configured, and sends authentication requests.
2. The AAA server authenticates users.

   The AAA server completes user authentication and sends the result to the BRAS. The security server, if configured, completes security authorization and sends the result to the BRAS.
3. (Optional.) DHCP allocates IP addresses.

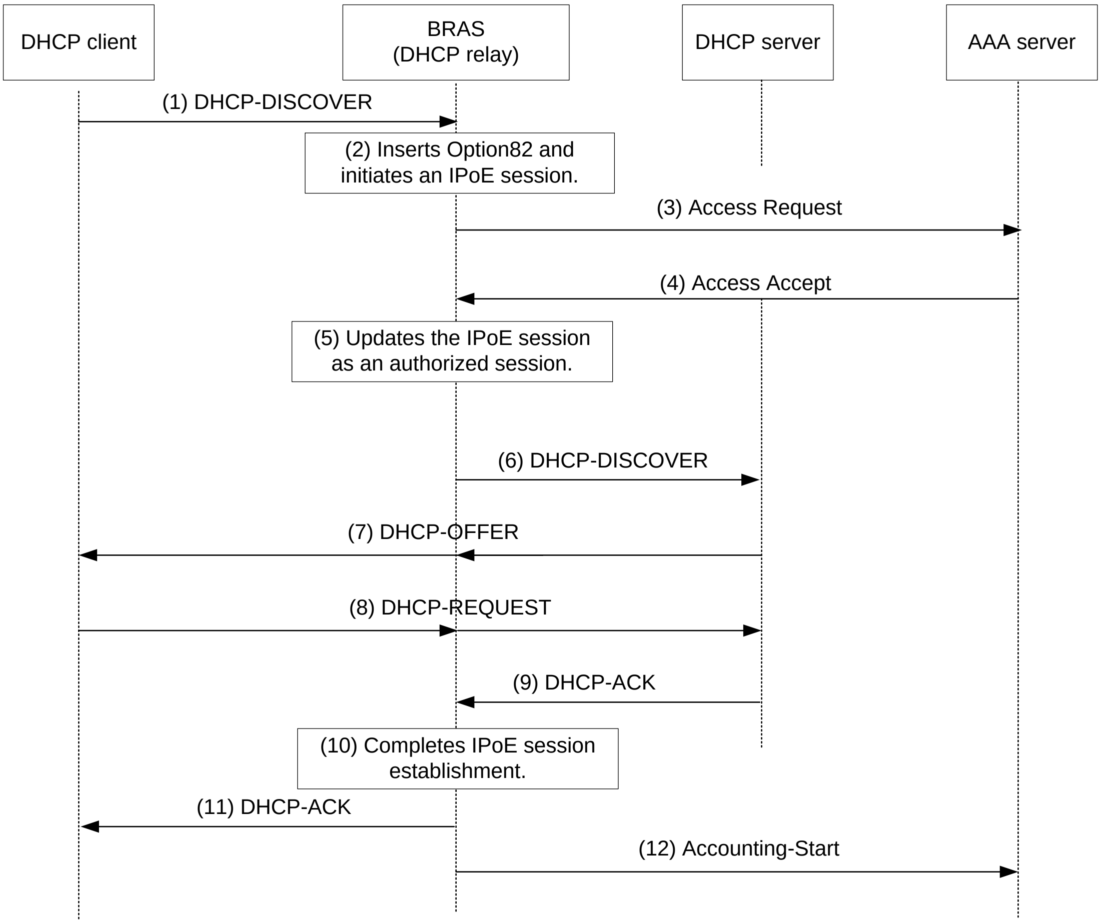
   The DHCP server assigns an IP address to a DHCP user and IPoE assigns an IPv6 prefix to an IPv6-ND-RS user.
4. The BRAS performs access control.

   The BRAS permits the user to get online and performs access control and billing based on the authorized result.

Access procedure for DHCP users

This section uses a DHCPv4 user as an example to illustrate the access procedure for DHCP users. The BRAS operates as a DHCP relay.

Figure 2 Access procedure for a DHCPv4 user



1. The DHCP client sends a DHCP-DISCOVER message to the BRAS.
2. The BRAS inserts Option 82 in the DHCP-DISCOVER message, and creates an IPoE session.
3. The BRAS sends the AAA server an access request that includes user information, such as the client ID and source MAC address.
4. The AAA server returns an access accept that contains authorization information to the BRAS if the authentication succeeds. If the authentication fails, the AAA server returns a reject message.
5. The BRAS marks the IPoE session state as success and forwards the DHCP-DISCOVER message to the DHCP server if the authentication succeeds. If the authentication fails, the BRAS marks the session as failure and discards the DHCP-DISCOVER message.
6. The DHCP server sends a DHCP-OFFER message to the BRAS.
7. The BRAS forwards the DHCP-OFFER message to the DHCP client.
8. The DHCP client sends a DHCP-REQUEST message to the BRAS.
9. The BRAS forwards the DHCP-REQUEST message to the specified DHCP sever.
10. The DHCP server sends a DHCP-ACK message containing the assigned IP address to the BRAS.
11. The BRAS performs the following:
    a. Obtains address information from the DHCP-ACK message.
    b. Assigns a user profile.
    c. Updates the IPoE session information.
    d. Forwards the DHCP-ACK message to the client.

   e. Marks the session state as online.

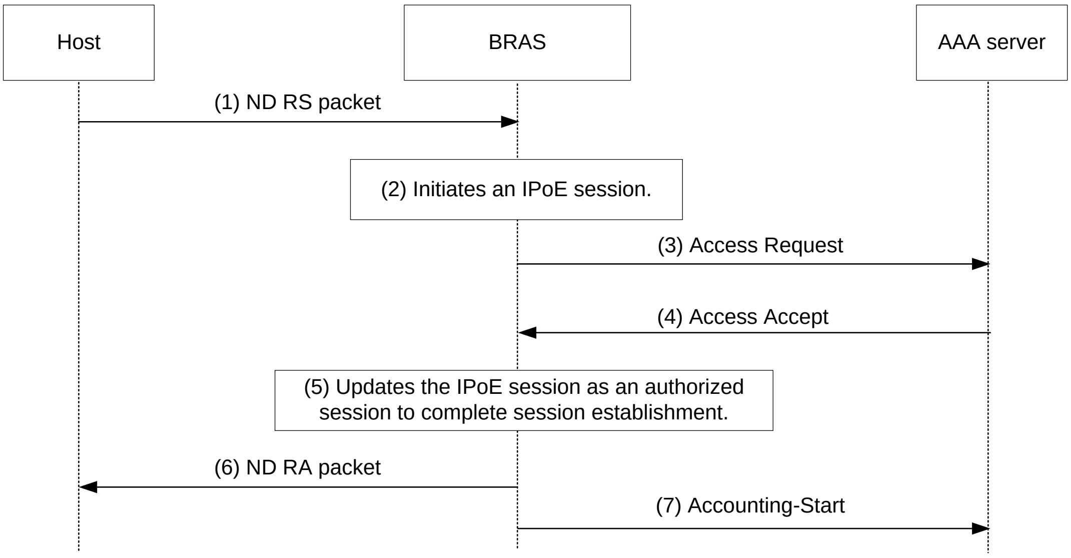
   If the authentication fails, the BRAS marks the session as failure and discards the DHCP-DISCOVER message.

12. The DHCP client obtains configuration information from the DHCP-ACK message.
13. The BRAS sends the AAA server a message to start accounting.

Access procedure for IPv6-ND-RS users

This example uses a Layer 2 device as the BRAS.

Figure 3 Access procedure for IPv6-ND-RS users

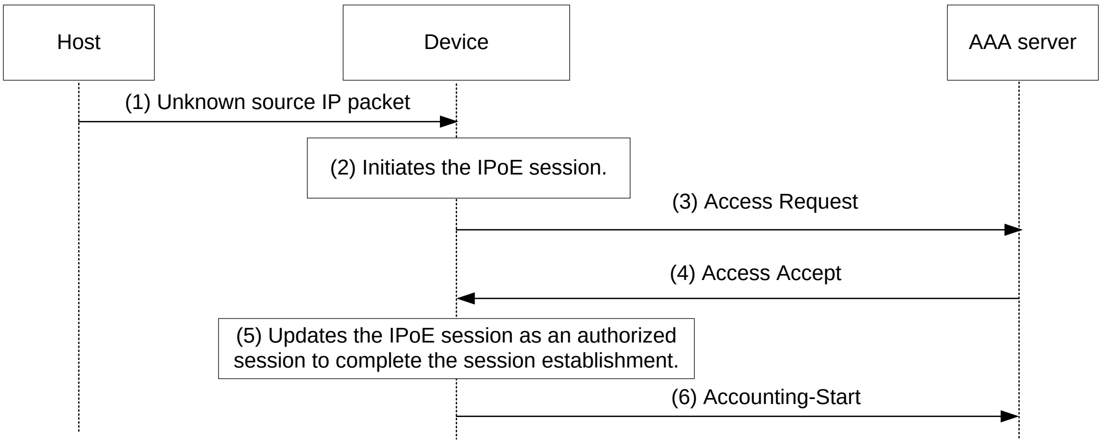

Host
BRAS
AAA server
(1) ND RS packet
(2) Initiates an IPoE session.
(3) Access Request
(4) Access Accept
(5) Updates the IPoE session as an authorized session to complete session establishment.
(6) ND RA packet
(7) Accounting-Start

1. The host sends an IPv6 ND RS packet to the BRAS.
2. The BRAS initiates an IPoE session and sends the AAA server an access request that contains user information, such as the source MAC address.
3. The AAA server returns an access accept that contains authorization information to the BRAS if the authentication succeeds. If the authentication fails, the AAA server returns a reject message.
4. The BRAS performs the following:
   a. Generates an IPv6 address based on the host's MAC address and the IPv6 prefix.
   b. Updates the IPoE session information.
   c. Marks the session as success.

   If the authentication fails, the BRAS marks the session as failure and discards the IPv6 ND RS packet.
5. The BRAS assigns a user profile and sends the host an IPv6 ND RA packet containing the IPv6 prefix.
6. The host generates an IPv6 address based on the received IPv6 prefix.
7. The BRAS sends the AAA server a message to start the service accounting.

Access procedure for unclassified-IP users

Figure 4 Access procedure for unclassified-IP users


Host
Device
AAA server
(1) Unknown source IP packet
(2) Initiates the IPoE session.
(3) Access Request
(4) Access Accept
(5) Updates the IPoE session as an authorized session to complete the session establishment.
(6) Accounting-Start

1. The host sends an IP packet to the BRAS.
2. The BRAS obtains user information from the IP packet, and matches the user information against existing IPoE sessions.
   - If no match is found, the BRAS initiates an IPoE session for the user. (This section uses this case as an example.)
   - If the information matches an authenticated session, the BRAS forwards the IP packet.
   - If the information matches an unauthenticated session, the BRAS discards the IP packet.
3. The BRAS sends the AAA server an access request containing the obtained information, such as the source IP address or source MAC address.
4. The AAA server returns an access accept that contains authorization information if the authentication succeeds. If the authentication fails, the AAA server returns a reject message.
5. The BRAS assigns a user profile and marks the IPoE session state as online.
6. The BRAS sends the AAA server a message to start the service accounting.

Access procedure for static and leased users

The access procedure for static users is as follows:

1. The user statically configures an IPoE session at the CLI on the BRAS.
2. The user sends an IP packet to the BRAS.
3. The BRAS obtains user information from the IP packet, and sends the AAA server an access request containing configured IPoE session information.
4. The AAA server returns an access accept that contains authorization information if the authentication succeeds. If the authentication fails, the AAA server returns a reject message.
5. The BRAS assigns the user profile and marks the IPoE session state as online.
6. The host receives the user profile.
7. The BRAS sends the AAA server a message to start the service accounting.

The access procedure for leased users is as follows:

1. The user statically configures an IPoE session at the CLI on the BRAS.
2. The BRAS automatically obtains configured IPoE session information and sends the AAA server an access request.
3. The AAA server returns an access accept that contains authorization information if the authentication succeeds. If the authentication fails, the AAA server returns a reject message.
4. The BRAS assigns the user profile and marks the IPoE session state as online.

5. The host receives the user profile.
6. The BRAS sends the AAA server a message to start the service accounting.

Support for MPLS L3VPN

IPoE supports MPLS L3VPN. It uses AAA to authorize VPNs for users. Before you bind a VPN instance to an interface, you must delete existing IPoE sessions on the interface for the users to communicate in their authorized VPNs.

NOTE:

Leased users do not support AAA-authorized VPNs through ISP domains or AAA servers. For more information about VPN authorization through ISP domains, see *Security Configuration Guide*.

Support for ITA

ITA provides accounting and bandwidth solutions for users based on the destination addresses they access.

For more information about configuring ITA, see *Security Configuration Guide*.

Restrictions and guidelines: IPoE configuration

IPoE supports the following interfaces:

- Layer 3 aggregate interfaces.
- Layer 3 aggregate subinterfaces
- Layer 3 Ethernet interfaces.
- Layer 3 Ethernet subinterfaces.

IPoE tasks at a glance

To configure IPoE, perform the following tasks:

1. Enabling IPoE and setting the IPoE access mode
2. Configure user types
   - Configuring dynamic individual users
   - Configuring static individual users
   - Configuring leased users

   Individual users and leased users cannot be configured on the same interface. Dynamic and static individual users can be configured on the same interface.
3. (Optional.) Configuring service-specific ISP domains
4. (Optional.) Configuring the quiet timer for users
5. (Optional.) Configuring online detection for IPoE users
6. (Optional.) Configuring NAS-Port-Type for an interface
7. (Optional.) Enabling IPoE user logging
8. (Optional.) Configuring NAS-Port-ID formats

Prerequisites for IPoE

Complete the following configuration as required:

- Configure the DHCP server.
- Enable the DHCP relay agent on the BRAS.
- Configure the RADIUS server and client. For more information about how to configure a RADIUS client, see AAA configuration in *Security Configuration Guide*.
- Configure security policies on the iMC security server and configure the security server's IP address on the BRAS. For more information about how to configure a security server, see AAA configuration in *Security Configuration Guide*.
- Configure local user accounts on the BRAS if local authentication is used. For more information about how to configure a local user account, see AAA configuration in *Security Configuration Guide*.
- Make sure the hosts, BRAS, and servers can reach each other.

Enabling IPoE and setting the IPoE access mode

Restrictions and guidelines

To change the IPoE access mode, disable IPoE, and then set the new IPoE mode when you enable IPoE.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Enable IPoE and set the IPoE access mode on an IPv4 interface
   ip subscriber { **l2-connected** | **routed** } **enable**
   By default, IPv4 IPoE is disabled.
4. Enable IPoE and set the IPoE access mode on an IPv6 interface
   ipv6 subscriber { **l2-connected** | **routed** } **enable**
   By default, IPv6 IPoE is disabled.

Configuring dynamic individual users

Dynamic individual user tasks at a glance

To configure dynamic individual users, perform the following tasks:

1. Enabling dynamic individual users
2. (Optional.) Configuring authentication user naming conventions for dynamic individual users
3. (Optional.) Configuring passwords for dynamic individual users
4. (Optional.) Configuring ISP domains for dynamic individual users
5. (Optional.) Configuring the maximum number of dynamic IPoE sessions
6. (Optional.) Configuring trusted DHCP options for DHCP users

Enabling dynamic individual users

About this task

Dynamic individual users include the unclassified-IP user, IPv6-ND-RS user, and DHCP user. After IPoE is enabled on an interface, the BRAS discards packets from users by default. You must enable dynamic individual users on the interface to trigger IPoE session establishment. You can enable multiple dynamic individual users on an interface.

Restrictions and guidelines

It requires the BRAS to send IPv6 ND RA packets. The interval for sending IPv6 ND RA packets should be no less than 6 minutes.

The IPv6-ND-RS user supports only hosts that use layer-2 access mode.

As a best practice, configure both the unclassified-IP user and IPv6-ND-RS user for an IPv6 interface. PCs running Windows generate IPv6 addresses randomly or using the EUI-64 method. The unclassified-IP user supports packets with randomly-generated IPv6 addresses. The IPv6-ND-RS user supports packets with EUI-64-generated IPv6 addresses.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the IPv4 dynamic individual user.

   ip subscriber initiator { **dhcp** | **unclassified-ip** } **enable**

   By default, no IPv4 dynamic individual users are enabled.

4. Enable the IPv6 dynamic individual user:

   ipv6 subscriber initiator { **dhcp** | **ndrs** | **unclassified-ip** } **enable**

   By default, no IPv6 dynamic individual users are enabled.

Configuring authentication user naming conventions for dynamic individual users

About this task

Usernames configured for dynamic individual users must be the same as those configured on the AAA server.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure authentication user naming conventions for IPv4 dynamic individual users.

   - Configure an authentication user naming convention for DHCPv4 users.

     ip subscriber dhcp username include { **circuit-id** [**separator** *separator*] | **client-id** [**separator** *separator*] | **nas-port-id** [**separator** *separator*] | **port** [**separator** *separator*] | **remote-id** [**separator** *separator*] | **second-vlan** [**separator** *separator*] | **slot** [**separator** *separator*] | **source-mac** [**address-separator** *address-separator*]

```
[separator separator ]|subslot [separator separator ]|sysname
[separator separator]|vendor-class [separator separator ] |
vendor-specific [separator separator ] | vlan [separator separator ]}
*
```

By default, usernames for DHCPv4 users are source MAC addresses.

- Configure an authentication user naming convention for unclassified-IP users.

```
ip subscriber unclassified-ip username include { nas-port-id
[separator separator ] | port [separator separator ]|second-vlan
[separator separator ]| slot [separator separator ]|source-ip
[ address-separator address-separator ] [separator separator ]|
source-mac [address-separator address-separator ] [separator
separator ]|subslot [separator separator]|sysname [separator
separator ]| vlan [separator separator ] }*
```

By default, usernames for unclassified-IP users are source IP addresses.

4. Configure authentication user naming conventions for IPv6 dynamic individual users.

- Configure an authentication user naming convention for DHCPv6 users.

```
ipv6 subscriber dhcp username include{ circuit-id [separator
separator ] | client-id [separator separator ]|nas-port-id
[separator separator ] | port [separator separator ]| remote-id
[separator separator ] | second-vlan [separator separator]| slot
[separator separator]| source-mac [ address-separator
address-separator ] [separator separator ] |subslot [separator
separator ]|sysname [separator separator ]|vendor-class [separator
separator ] | vendor-specific [separator separator ] | vlan [separator
separator ]}*
```

By default, usernames for DHCPv6 are source MAC addresses.

- Configure an authentication user naming convention for IPv6-ND-RS users.

```
ipv6 subscriber ndrs username include { nas-port-id [separator
separator ] | port [separator separator ]| second-vlan [separator
separator ]| slot [separator separator ] | source-mac
[ address-separator address-separator ] [separator separator ] |
subslot [separator separator ]|sysname [separator separator ]| vlan
[separator separator ]}*
```

By default, usernames for IPv6-ND-RS users are source MAC addresses.

- Configure an authentication user naming convention for unclassified-IP users.

```
ipv6 subscriber unclassified-ip username include{ nas-port-id
[ separator separator ] | port [separator separator ] |second-vlan
[separator separator ]|slot [separator separator ]|source-ip
[ address-separator address-separator ] [separator separator ] |
source-mac [ address-separator address-separator ] [separator
separator ] |subslot [separator separator]|sysname [separator
separator ]| vlan [separator separator]}*
```

By default, usernames for unclassified-IP users are source IP addresses.

Configuring passwords for dynamic individual users

About this task

Passwords configured for dynamic individual users must be the same as those configured on the AAA server.

If you configure multiple passwords for an DHCP user, the passwords are used in the following order:

1. Password specified in trusted Option 60 or Option 16.
2. Password specified the **ip subscriber password** or **ipv6 subscriber password** command.
3. Default system password.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a password for IPv4 IPoE users.
   - Configure a password for IPv4 dynamic individual users.

     ip subscriber password { **ciphertext** | **plaintext** } *string*

     The default password for dynamic individual users is **vlan**.

   - Specify a string from the Option 60 as the password for IPv4 DHCP users.

     ip subscriber dhcp password option60 [**offset** *offset*] [**length** *length*]

     By default, the BRAS does not use the password specified in Option 60 for DHCP users.

     Configure Option 60 as the trusted DHCP option for the password specified by this command to take effect. For more information about Option 60, see "Configuring trusted DHCP options for DHCP users."

4. Configure a password for IPv6 IPoE users.
   - Configure a password for IPv6 dynamic individual users.

     ipv6 subscriber password { **ciphertext** | **plaintext** } *string*

     The default password for dynamic individual users is **vlan**.

   - Specify a string from the Option 16 as the password for IPv6 dynamic individual users:

     ipv6 subscriber dhcp password option16 [**offset** *offset*] [**length** *length*]

     By default, the BRAS does not use the password specified in Option 16 for DHCP users.

     Configure DHCPv6 Option 16 as the trusted DHCP option for the password specified by this command to take effect. For more information about Option 16, see "Configuring trusted DHCP options for DHCP users."

Configuring ISP domains for dynamic individual users

About this task

The following table shows how the BRAS selects ISP domains for dynamic individual users.

Dynamic individual users	Order in selecting an ISP domain
DHCP user	• ISP domain automatically selected from DHCPv4 Option 60 or DHCPv6 Option 16 if the option is trusted • Interface-specific ISP domain • Default system ISP domain
IPv6-ND-RS user	• Interface-specific ISP domain • Default system ISP domain

Dynamic individual users	Order in selecting an ISP domain
Unclassified-IP user	• Service-specific ISP domain • Interface-specific ISP domain • Default system ISP domain

For more information about how to configure service-specific ISP domains, see "Configuring service-specific ISP domains."

For more information about how to configure the default system ISP domain, see AAA configuration in *Security Configuration Guide*.

Restrictions and guidelines

Configure trusted DHCP options before you configure ISP domains automatically selected from DHCPv4 Option 60 or DHCPv6 Option 16. For more information about how to configure trusted DHCP options, see "Configuring trusted DHCP options for DHCP users."

The specified ISP domain must exist on the BRAS.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Configure an ISP domain for IPv4 dynamic individual users.
   ip subscriber { **dhcp** | **unclassified-ip** } **domain** *domain-name*
   By default, IPv4 dynamic individual users use the default system ISP domains.
4. Configure an ISP domain for IPv6 dynamic individual users.
   ipv6 subscriber { **dhcp** | **ndrs** | **unclassified-ip** } **domain** *domain-name*
   By default, IPv6 dynamic individual users use the default system ISP domains.

Configuring the maximum number of dynamic IPoE sessions

About this task

This feature limits the total number of dynamic IPoE sessions on an interface.

Restrictions and guidelines

You can set a smaller value than the number of existing dynamic IPoE sessions on an interface. In this scenario, the existing dynamic IPoE sessions are not affected.

Make sure the total maximum number of dynamic IPoE sessions for all interfaces on the device is less than the upper limit of the device. The maximum number of dynamic IPoE sessions supported by a device varies by license and device model. If the number of dynamic IPoE sessions reaches the upper limit on an interface or reaches the upper limit supported by the device, you cannot establish any additional IPoE sessions on the interface or on the device.

You can install a license that supports less dynamic IPoE sessions than the existing dynamic IPoE sessions. In this scenario, the existing dynamic IPoE sessions are not affected.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.

interface *interface-type interface-number*

3. Configure the maximum number of IPv4 dynamic IPoE sessions.

   ip subscriber { **dhcp** | **unclassified-ip** } **max-session** *max-number*

   By default, the maximum number of dynamic IPv4 IPoE sessions is not configured.

4. Configure the maximum number of IPv6 dynamic IPoE sessions.

   ipv6 subscriber { **dhcp** | **ndrs** | **unclassified-ip** } **max-session** *max-number*

   By default, the maximum number of dynamic IPv6 IPoE sessions is not configured.

Configuring trusted DHCP options for DHCP users

About this task

This feature enables a BRAS to obtain user access information from trusted DHCP options when the BRAS acts as a DHCP relay. The BRAS includes the obtained user access information in the RADIUS attributes sent to the RADIUS server based on the following matrix.

Table 1 Associated DHCP options for RADIUS attributes

RADIUS attributes	Associated DHCP options
NAS-PORT-ID	• DHCPv4 Option 82 Suboption Circuit-ID • DHCPv6 Option 18
DSL_AGENT_CIRCUIT_ID	• DHCPv4 Option 82 Suboption Circuit-ID • DHCPv6 Option 18
DSL_AGENT_REMOTE_ID	• DHCPv4 Option 82 Suboption Remote-ID • DHCPv6 Option 37

If the BRAS trusts DHCPv4 Option 60 and DHCPv6 Option 16, IPoE can use the ISP domains specified in the options when certain conditions exist. For more information about selecting ISP domains, see "Configuring ISP domains for dynamic individual users."

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure trusted DHCP options for DHCPv4 users.

   ip subscriber trust { **option60** | **option82** }

   By default, the BRAS does not trust DHCPv4 options.

4. Configure trusted DHCP options for DHCPv6 users.

   ipv6 subscriber trust { **option16** | **option18** | **option37** }

   By default, the BRAS does not trust DHCPv6 options.

Configuring static individual users

Static individual user tasks at a glance

To configure static individual users, perform the following tasks:

1. Enabling static individual users

2. Configuring static IPoE sessions
3. (Optional.) Configuring authentication user naming conventions for static individual users
4. (Optional.) Configuring passwords for static individual users
5. (Optional.) Configuring ISP domains for static individual users
6. (Optional.) Configuring the static IPoE whitelist feature

Enabling static individual users

About this task

This feature enables configured static IPoE sessions information to match IP packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IPv4 static individual users.

   ip subscriber initiator unclassified-ip enable

   By default, no IPv4 static individual users are enabled.

4. Enable IPv6 static individual users.

   ipv6 subscriber initiator unclassified-ip enable

   By default, no static individual users are enabled.

Configuring static IPoE sessions

About this task

Static individual users trigger IPoE session establishment by sending IP packets. If an IP packet matches a manually configured IPoE session, the BRAS authenticates the user and establishes an IPoE session.

Restrictions and guidelines

On one interface, a maximum of one static IPoE session can be configured for one IP address.

Configuration procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a static IPv4 IPoE session.

   ip subscriber session static ip *ip-address* [**vlan** *vlan-id* [**second-vlan** *vlan-id*]] [**mac** *mac-address*] [**domain** *domain-name*] [**description** *string*]

   By default, no static IPv4 IPoE session is configured.

   Only subinterfaces support parameters **vlan** and **second-vlan**.

4. Configure a static IPv6 IPoE session.

   ipv6 subscriber session static ipv6 *ipv6-address* [**vlan** *vlan-id* [**second-vlan** *vlan-id*]] [**mac** *mac-address*] [**domain** *domain-name*] [**description** *string*]

By default, no static IPv6 IPoE session is configured.

Only subinterfaces support parameters **vlan** and **second-vlan**.

Configuring authentication user naming conventions for static individual users

About this task

Usernames configured for static individual users must be the same as those configured on the AAA server.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an authentication user naming convention for IPv4 static individual users.

   ip subscriber unclassified-ip username include { **nas-port-id** [**separator** *separator*] | **port** [**separator** *separator*] | **second-vlan** [**separator** *separator*] | **slot** [**separator** *separator*] | **source-ip** [**address-separator** *address-separator*] [**separator** *separator*] | **source-mac** [**address-separator** *address-separator*] [**separator** *separator*] | **subslot** [**separator** *separator*] | **sysname** [**separator** *separator*] | **vlan** [**separator** *separator*] } *

   The default username is the source IPv4 address of packets sent by users.

4. Configure an authentication user naming convention for IPv6 static individual users.

   ipv6 subscriber unclassified-ip username include { **nas-port-id** [**separator** *separator*] | **port** [**separator** *separator*] | **second-vlan** [**separator** *separator*] | **slot** [**separator** *separator*] | **source-ip** [**address-separator** *address-separator*] [**separator** *separator*] | **source-mac** [**address-separator** *address-separator*] [**separator** *separator*] | **subslot** [**separator** *separator*] | **sysname** [**separator** *separator*] | **vlan** [**separator** *separator*] } *

   The default username is the source IPv6 address of packets sent by users.

Configuring passwords for static individual users

About this task

Passwords configured for static individual users must be the same as those configured on the AAA server.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a password for IPv4 static individual users.

   ip subscriber password { **ciphertext** | **plaintext** } *string*

   The default password for an IPv4 static individual user is **vlan**.

4. Configure a password for IPv6 static individual users.

   ipv6 subscriber password { **ciphertext** | **plaintext** } *string*

   The default password for an IPv6 static individual user is **vlan**.

Configuring ISP domains for static individual users

About this task

The BRAS selects ISP domains for static individual users in the following order:

- Domain configured in the static IPoE session
- Service-specific domain
- Interface-specific domain
- Default system domain

For more information about how to configure service-specific ISP domains, see "Configuring service-specific ISP domains." For more information about how to configure the default system domain, see *Security Configuration Guide*.

Restrictions and guidelines

The specified ISP domain must exist on the BRAS.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an ISP domain for static IPv4 individual users.

   ip subscriber unclassified-ip domain *domain-name*

   By default, static IPv4 individual users use the default system domain.

4. Configure an ISP domain for static IPv6 individual users.

   ipv6 subscriber unclassified-ip domain *domain-name*

   By default, static IPv6 individual users use the default system domain.

Configuring the static IPoE whitelist feature

About this task

With this feature enabled, only IPv4 or IPv6 traffic matching static IPv4 or IPv6 IPoE sessions can initiate IPoE authentication, and IPoE directly permits the other traffic without any processing.

In some scenarios, an interface might need to have both IPoE and portal authentication enabled. For example, both dumb terminals and broadband dial-up users exist on an interface. Dumb terminals (for example, monitoring cameras) need to come online through IPoE without portal authentication, and broadband dial-up users need to come online through portal Web authentication. In this case, you can enable the IPv4 or IPv6 IPoE whitelist feature on the interface.

When both the IPv4 or IPv6 IPoE whitelist feature and portal authentication are enabled on an interface, the following rules apply:

- If the IPv4 or IPv6 traffic of a user matches a static IPv4 or IPv6 IPoE session, the user is processed by the static IPv4 or IPv6 IPoE authentication flow. For an IPoE user to bypass authentication, specify the authentication and authorization modes as **none** in the ISP domain of the IPoE user.

- If the IPv4 or IPv6 traffic of a user does not match any IPv4 or IPv6 IPoE session, the user is processed by portal authentication.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the IPv4 IPoE whitelist feature.

   ip subscriber whitelist enable

   By default, the IPv4 IPoE whitelist is disabled.

4. Enable the IPv6 IPoE whitelist feature.

   ipv6 subscriber whitelist enable

   By default, the IPv6 IPoE whitelist is disabled.

Configuring leased users

Leased user tasks at a glance

To configure leased users, perform the following tasks:

1. Configuring leased users
   - Configuring interface-leased users
   - Configuring subnet-leased users

   Interface-leased users and subnet-leased users cannot be configured on the same interface.

2. Configuring ISP domains for leased users

Configuring interface-leased users

About this task

An interface-leased user represents hosts that rent the same interface.

Restrictions and guidelines

You can configure up to one IPv4 interface-leased user and one IPv6 interface-leased user on an interface.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an IPv4 interface-leased user.

   ip subscriber interface-leased username *name* **password** { **ciphertext** | **plaintext** } *string* [**domain** *domain-name*]

   By default, no IPv4 interface-leased user is configured.

4. Configure an IPv6 interface-leased user.

   ipv6 subscriber interface-leased username *name* **password** { **ciphertext** | **plaintext** } *string* [**domain** *domain-name*]

By default, no IPv6 interface-leased user is configured.

Configuring subnet-leased users

About this task

A subnet-leased user represents hosts that rent a subnet of an interface.

Restrictions and guidelines

You can configure multiple subnet-leased users on an interface. Different subnets must have the same mask length. Each subnet can be bound to only one subnet-leased user.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an IPv4 subnet-leased user.

   ip subscriber subnet-leased ip *ip-address* { *mask* | *mask-length* } **username** *name* **password** { **ciphertext** | **plaintext** } *string* [**domain** *domain-name*]

   By default, no IPv4 subnet-leased user is configured.

4. Configure an IPv6 subnet-leased user.

   ipv6 subscriber subnet-leased ipv6 *ipv6-address prefix-length* **username** *name* **password** { **ciphertext** | **plaintext** } *string* [**domain** *domain-name*]

   By default, no IPv6 subnet-leased user is configured.

Configuring ISP domains for leased users

About this task

The BRAS selects ISP domains for leased users in the following order:

- Domains configured for each interface-leased user and each subnet-leased user
- Service-specific domain
- Domain configured for leased users
- Default system domain

For more information about how to configure the service-specific ISP domains, see "Configuring service-specific ISP domains."

For more information about how to configure the default system domain, see AAA configuration in *Security Configuration Guide*.

Restrictions and guidelines

The specified ISP domain must exist on the BRAS.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a domain collectively for IPv4 leased users.

ip subscriber unclassified-ip domain *domain-name*

By default, IPv4 leased users use the default system ISP domain.

4. Configure a domain collectively for IPv6 leased users.

   ipv6 subscriber unclassified-ip domain *domain-name*

   By default, IPv6 leased users use the default system ISP domain.

Configuring service-specific ISP domains

About configuring service-specific ISP domains

This task enables you to assign ISP domains to users based on services. You can classify services by VLAN ID, 802.1P, and DSCP carried in packets from users.

Restrictions and guidelines

For DHCPv4 users, the trusted Option 60 configuration takes precedence over the global service identifier configuration.

For DHCPv6 users, the trusted Option 16 configuration takes precedence over the global service identifier configuration.

You must specify an identifier for a service before you bind an ISP domain to the service. Otherwise, the binding does not take effect.

Configuring service-specific ISP domains for IPv4 users

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a service identifier for IPv4 users.

   ip subscriber service-identify { **8021p** { **second-vlan** | **vlan** } | **dscp** | **second-vlan** | **vlan** }

   By default, no service identifier is configured for IPv4 users.

4. Configure service-specific ISP domains for IPv4 users.

   - Bind an ISP domain to a VLAN list.

     ip subscriber vlan *vlan-list* **domain** *domain-name*

   - Bind an ISP domain to an 802.1p list.

     ip subscriber 8021p *8021p-list* **domain** *domain-name*

   - Bind an ISP domain to a DSCP list.

     ip subscriber dscp *dscp-value-list* **domain** *domain-name*

   By default, no service-specific ISP domains are configured for IPv6 users.

Configuring service-specific ISP domains for IPv6 users

1. Enter system view.

   system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Configure a service identifier for IPv6 users.

   ipv6 subscriber service-identify { **8021p** { **second-vlan** | **vlan** } | **dscp** | **second-vlan** | **vlan** }

   By default, no service identifier is configured for IPv6 users.

4. Configure service-specific ISP domains for IPv6 users.
   - Bind an ISP domain to a VLAN list.

     ipv6 subscriber vlan *vlan-list* **domain** *domain-name*
   - Bind an ISP domain to an 802.1p list.

     ipv6 subscriber 8021p *8021p-list* **domain** *domain-name*
   - Bind an ISP domain to a DSCP list.

     ipv6 subscriber dscp *dscp-value-list* **domain** *domain-name*

   By default, no service-specific ISP domains are configured for IPv6 users.

Configuring the quiet timer for users

About this task

If this feature is enabled, the quiet timer starts when number of consecutive authentication failures of a user reaches the limit in the specified period. During the quiet timer period, packets from the user are dropped. After the quiet timer expires, the BRAS performs authentication upon receiving a packet from the user.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the quiet timer for IPv4 IPoE users.

   ip subscriber timer quiet *time*

   By default, the quite timer is disabled for IPv4 IPoE users.

4. Configure the quiet timer for IPv6 IPoE users.

   ipv6 subscriber timer quiet *time*

   By default, the quite timer is disabled for IPv6 IPoE users.

Configuring online detection for IPoE users

About this task

Online detection enables the BRAS to periodically detect the status of a user. It uses ARP or ICMP requests to detect IPv4 users, and uses NS packets of the ND protocol or ICMPv6 requests to detect IPv6 users.

After you configure online detection, the BRAS starts a detection timer to detect online users. If the BRAS does not receive user packets from a user when the detection timer expires, it sends a detection packet to the user and performs the following operations:

- If the BRAS receives user packets within the maximum number of detection attempts, the BRAS assumes that the user is online. It resets the detection timer, and starts the next detection attempt.

- If the BRAS does not receive user packets within the maximum number of detection attempts, the BRAS assumes that the user is offline and deletes the user session.

Restrictions and guidelines

This feature supports only individual users.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure online detection for IPv4 IPoE users.

   ip subscriber user-detect { **arp** | **icmp** } **retry** *retries* **interval** *interval*

   By default, online detection is disabled for IPv4 IPoE users.

4. Configure online detection for IPv6 IPoE users.

   ipv6 subscriber user-detect { **icmpv6** | **nd** } **retry** *retries* **interval** *interval*

   By default, online detection is disabled for IPv6 IPoE users.

Configuring NAS-Port-Type for an interface

About this task

The NAS-Port-Type attribute carries information about the access interface. The BRAS includes the configured NAS-Port-Type in RADIUS requests sent to the RADIUS server.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the IPv4 NAS-Port-Type.

   ip subscriber nas-port-type { **802.11** | **adsl-cap** | **adsl-dmt** | **async** | **cable** | **ethernet** | **g.3-fax** | **hdlc** | **idsl** | **isdn-async-v110** | **isdn-async-v120** | **isdn-sync** | **piafs** | **sdsl** | **sync** | **virtual** | **wireless-other** | **x.25** | **x.75** | **xdsl** }

   The default IPv4 NAS-Port-Type is **Ethernet**.

4. Configure the IPv6 NAS-Port-Type.

   ipv6 subscriber nas-port-type { **802.11** | **adsl-cap** | **adsl-dmt** | **async** | **cable** | **ethernet** | **g.3-fax** | **hdlc** | **idsl** | **isdn-async-v110** | **isdn-async-v120** | **isdn-sync** | **piafs** | **sdsl** | **sync** | **virtual** | **wireless-other** | **x.25** | **x.75** | **xdsl** }

   The default IPv6 NAS-Port-Type is **Ethernet**.

Enabling IPoE user logging

About this task

The IPoE user logging feature enables the device to generate IPoE logs and send them to the information center. Logs are generated after a user comes online successfully, fails to come online, normally goes offline, or abnormally goes offline. A log entry contains information such as the username, IP address, interface name, inner VLAN, outer VLAN, MAC address, and failure causes. For information about the log destination and output rule configuration in the information center, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

Typically, disable this feature to prevent excessive IPoE log output.

Procedure

1. Enter system view.

   system-view

2. Enable IPv4 IPoE user logging.

   ip subscriber access-user log enable [**successful-login** | **failed-login** | **logout** [**normal**] [**abnormal**]] *

   By default, IPv4 IPoE user logging is disabled.

3. Enable IPv6 IPoE user logging.

   ipv6 subscriber access-user log enable [**successful-login** | **failed-login** | **logout** [**normal**] [**abnormal**]] *

   By default, IPv6 IPoE user logging is disabled.

Configuring NAS-Port-ID formats

About configuring NAS-Port-ID formats

The NAS-Port-ID RADIUS attribute specifies access location of a user. The BRAS supports the following formats for NAS-Port-ID:

- **version 1.0**—Format for China Telecom.
- **version 2.0**—Format specified in *YDT 2275-2011 Subscriber Access Loop (Port) Identification in Broadband Access Networks*.

You can configure the following settings if version 2.0 is used when the BRAS acts as a DHCP relay:

- Configure DHCPv4 Option 82 or DHCPv6 Option 18 as a trusted DHCP option and obtain information from the trusted option.
- Include the NAS information and obtained option information in NAS-Port-ID.

Configuring the NAS-Port-ID format for IPv4 users

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the NAS-Port-ID format for IPv4 users.

   ip subscriber nas-port-id format cn-telecom { **version1.0** | **version2.0** }

   The default format is **version1.0**.

4. (Optional.) Configure the trusted DHCPv4 option 82 for IPv4 users.

   ip subscriber trust option82

   By default, the BRAS does not trust Option 82.

5. (Optional.) Include the NAS information and DHCPv4 option 82 information in NAS-Port-ID for IPv4 users.

ip subscriber nas-port-id nasinfo-insert

By default, the BRAS includes only information obtained from the trusted DHCPv4 option 82 in NAS-Port-ID.

Configuring the NAS-Port-ID format for IPv6 users

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Configure the NAS-Port-ID format for IPv6 users.
   ipv6 subscriber nas-port-id format cn-telecom { **version1.0** | **version2.0** }
   The default format is **version1.0**.
4. (Optional.) Configure the trusted DHCPv6 option 18 for IPv6 users.
   ipv6 subscriber trust option18
   By default, the BRAS does not trust Option 18.
5. (Optional.) Include the NAS information and DHCPv6 option 18 information in NAS-Port-ID for IPv6 users.
   ipv6 subscriber nas-port-id nasinfo-insert
   By default, the BRAS includes only information obtained from the trusted DHCPv6 option 18 in NAS-Port-ID.

Display and maintenance commands for IPoE

Display and maintenance commands for IPv4 IPoE

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPoE session information for individual users.	**display ip subscriber session** [**interface** *interface-type interface-number*] [**domain** *domain-name* | **ip** *ip-address* [**vpn-instance** *vpn-instance-name*] | **mac** *mac-address* | **static** | **username** *name*] [**slot** *slot-number*] [**verbose**]
Display information about IPoE interface-leased users.	**display ip subscriber interface-leased** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display information about IPoE subnet-leased users.	**display ip subscriber subnet-leased** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPoE session statistics for individual users.	**display ip subscriber session statistics** [**session-type** { **dhcp** | **static** | **unclassified-ip** }] [**interface** *interface-type interface-number*] [**slot** *slot-number*]

Task	Command
Display IPoE session statistics for interface-leased users.	**display ip subscriber interface-leased statistics** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPoE session statistics for subnet-leased users.	**display ip subscriber subnet-leased statistics** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display offline statistics for users.	**display ip subscriber offline statistics** [**interface** *interface-type interface-number*]
Delete dynamic IPoE sessions and log out the users.	**reset ip subscriber session** [**interface** *interface-type interface-number*] [**domain** *domain-name* | **ip** *ip-address* [**vpn-instance** *vpn-instance-name*] | **mac** *mac-address* | **username** *name*]
Delete offline statistics for users.	**reset ip subscriber offline statistics** [**interface** *interface-type interface-number*]

Display and maintenance commands for IPv6 IPoE

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPoE session information for individual users.	**display ipv6 subscriber session** [**interface** *interface-type interface-number*] [**domain** *domain-name* | **ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*] | **mac** *mac-address* | **static** | **username** *name*] [**slot** *slot-number*] [**verbose**]
Display information about IPoE interface-leased users.	**display ipv6 subscriber interface-leased** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display information about IPoE subnet-leased users.	**display ipv6 subscriber subnet-leased** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPoE session statistics for individual users.	**display ipv6 subscriber session statistics** [**session-type** { **dhcp** | **ndrs** | **static** | **unclassified-ip** }] [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPoE session statistics for interface-leased users.	**display ipv6 subscriber interface-leased statistics** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Display IPoE session statistics for subnet-leased users.	**display ipv6 subscriber subnet-leased statistics** [**interface** *interface-type interface-number*] [**slot** *slot-number*]

Task	Command
Display offline statistics for users.	**display ipv6 subscriber offline statistics** [**interface** *interface-type interface-number*]
Delete dynamic IPoE sessions and log out the users.	**reset ipv6 subscriber session** [**interface** *interface-type interface-number*] [**domain** *domain-name* | **ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*] | **mac** *mac-address* | **username** *name*]
Delete offline statistics for users.	**reset ipv6 subscriber offline statistics** [**interface** *interface-type interface-number*]

Troubleshooting IPoE

DHCP clients failed to come online

Symptom

DHCP clients cannot come online, although network connections and interface IPoE configurations are correct.

Solution

To solve the problem:

1. Use debug commands or a packet analyzer to check DHCP packets from the DHCP client.

   By default, DHCPv4 and DHCPv6 clients use ISP domains specified in Option 60 and Option 16, respectively.
2. If the DHCPv4 packet carries Option 60 or the DHCPv6 packet carries Option 16, verify that the ISP domain in the option exists on the BRAS.
3. If the DHCP packet does not carry Option 60 or Option 16, verify that the ISP domain specified on the interface exists on the BRAS.
4. If the problem persists, contact NSFOCUS Support.

Contents

Managing public keys

About public key management

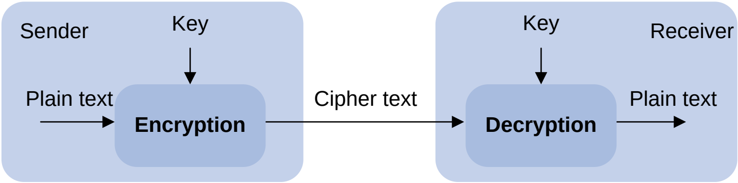
This chapter describes public key management for the following asymmetric key algorithms:

- Revest-Shamir-Adleman Algorithm (RSA).
- Digital Signature Algorithm (DSA).
- Elliptic Curve Digital Signature Algorithm (ECDSA).
- SM2.

Asymmetric key algorithm overview

Asymmetric key algorithms are used by security applications to secure communications between two parties, as shown in Figure 1. Asymmetric key algorithms use two separate keys (one public and one private) for encryption and decryption. Symmetric key algorithms use only one key.

Figure 1 Encryption and decryption



A key owner can distribute the public key in plain text on the network but must keep the private key in privacy. It is mathematically infeasible to calculate the private key even if an attacker knows the algorithm and the public key.

Usage of asymmetric key algorithms

Security applications (such as SSH, SSL, and PKI) use the asymmetric key algorithms for the following purposes:

- **Encryption and decryption**—Any public key receiver can use the public key to encrypt information, but only the private key owner can decrypt the information.
- **Digital signature**—The key owner uses the private key to digitally sign information to be sent. The receiver decrypts the information with the sender's public key to verify information authenticity.

RSA, DSA, ECDSA, and SM2 can all perform digital signature, but only RSA can perform encryption and decryption.

Public key management tasks at a glance

To manage public keys, perform the following tasks:

1. Creating a local key pair
2. Importing a local key pair
3. Distributing a local host public key

   Choose one of the following tasks:

   - Exporting a host public key
   - Displaying a host public key

   To enable the peer device to authenticate the local device, you must distribute the local device's public key to the peer device.

4. Configuring a peer host public key

   Choose one of the following tasks:

   - Importing a peer host public key from a public key file
   - Entering a peer host public key

   To encrypt information sent to a peer device or authenticate the digital signature of the peer device, you must configure the peer device's public key on the local device.

5. (Optional.) Destroying a local key pair

Creating a local key pair

Restrictions and guidelines

When you create a local key pair, follow these guidelines:

- The key algorithm must be the same as required by the security application.
- When you create an RSA or DSA key pair, enter an appropriate key modulus length at the prompt. The longer the key modulus length, the higher the security, and the longer the key generation time.

  When you create an ECDSA key pair, choose the appropriate elliptic curve. The elliptic curve determines the ECDSA key length. The longer the key length, the higher the security, and the longer the key generation time.

  When you create an SM2 key pair, you do not need to specify the key length. Only a 256-bit SM2 key pair can be created.

  See Table 1 for more information about key modulus lengths and key lengths.
- If you do not assign the key pair a name, the system assigns the default name to the key pair and marks the key pair as **default**. You can also assign the default name to another key pair, but the system does not mark the key pair as **default**. The key pair name must be unique among all manually named key pairs that use the same key algorithm. If a name conflict occurs, the system asks whether you want to overwrite the existing key pair.
- The key pairs are automatically saved and can survive system reboots.

Table 1 A comparison of different types of asymmetric key algorithms

Type	Generated key pairs	Modulus/key length
RSA	• One host key pair, if you specify a key pair name. • One server key pair and one host key pair, if you do not specify a key pair name. Both key pairs use their default names. **NOTE:** Only SSH 1.5 uses the RSA server key pair.	Key modulus length: 512 to 2048 bits. Default: 1024 bits. To ensure security, use a minimum of 768 bits.
DSA	One host key pair.	Key modulus length: 512 to 2048 bits. Default: 1024 bits. To ensure security, use a minimum of 768 bits.
ECDSA	One host key pair.	Key length: 192, 256, 384, or 521 bits.

Type	Generated key pairs	Modulus/key length
SM2	One host key pair.	Key length: 256 bits.

Procedure

1. Enter system view.

   system-view

2. Create a local key pair.

   public-key local create { **dsa** | **ecdsa** [**secp192r1** | **secp256r1** | **secp384r1** | **secp521r1**] | **rsa** } [**name** *key-name*]

   public-key local create sm2 [**name** *key-name*] [**on** *device-name*]

Importing a local key pair

About this task

This task imports a key pair from a key pair file to the device. The imported key pairs are automatically saved and can survive system reboots.

Perform this task when the key pair to be imported is saved in a different file than the certificate. If the key pair and the certificate are saved in the same file, the device can obtain the key pair by importing the certificate.

Restrictions and guidelines

The device supports importing the RSA host key pair but not the RSA server key pair.

If you do not assign the key pair a name, the system assigns the default name to the key pair and marks the key pair as **default**. You can also assign the default name to another key pair, but the system does not mark the key pair as **default**. The name of a key pair must be unique among all manually named key pairs that use the same key algorithm. If a name conflict occurs, the system asks whether you want to overwrite the existing key pair.

To import the encrypted key pair into the device successfully, provide the decryption password.

See Table 2 for information about supported key modulus lengths and key lengths.

Table 2 Length of key pair

Type	Modulus/key length
RSA	Key modulus length: 512 to 2048 bits.
ECDSA	Key length: 192, 256, 384, or 521 bits.

Prerequisites

Before performing this task, save the key pair file to the local storage directory of the device through FTP or other methods.

Procedure

1. Enter system view.

   system-view

2. Import a local key pair.

   public-key local import { **ecdsa** | **rsa** } [*key-name*] **filename** *filename*

Distributing a local host public key

About distribution of local host public keys

You must distribute a local host public key to a peer device so the peer device can perform the following operations:

- Use the public key to encrypt information sent to the local device.
- Authenticate the digital signature signed by the local device.

To distribute a local host public key, you must first export or display the key.

- Export a host public key:
  - Export a host public key to a file.
  - Export a host public key to the monitor screen, and then save it to a file.

  After the key is exported to a file, transfer the file to the peer device. On the peer device, import the key from the file.
- Display a host public key.

  After the key is displayed, record the key, for example, copy it to an unformatted file. On the peer device, you must literally enter the key.

Exporting a host public key

Restrictions and guidelines

When you export a host public key, follow these restrictions and guidelines:

- If you specify a file name in the command, the command exports the key to the specified file.
- If you do not specify a file name, the command exports the key to the monitor screen. You must manually save the exported key to a file.

Procedure

1. Enter system view.

   system-view
2. Export a local host public key.
   - Export an RSA host public key:

     public-key local export rsa [**name** *key-name*] { **openssh** | **ssh1** | **ssh2** } [*filename*]
   - Export an ECDSA host public key.

     public-key local export ecdsa [**name** *key-name*] { **openssh** | **ssh2** } [*filename*]
   - Export a DSA host public key.

     public-key local export dsa [**name** *key-name*] { **openssh** | **ssh2** } [*filename*]
   - Export an SM2 host public key.

     public-key local export sm2 [**name** *key-name*] { **openssh** | **ssh2** } [*filename*]

Displaying a host public key

Perform the following tasks in any view:

- Display local RSA public keys.

  display public-key local rsa public [**name** *key-name*]

  Do not distribute the RSA server public key **serverkey (default)** to a peer device.
- Display local ECDSA public keys.

  display public-key local ecdsa public [**name** *key-name*]
- Display local DSA public keys.

  display public-key local dsa public [**name** *key-name*]
- Display local SM2 public keys.

  display public-key local sm2 public [**name** *key-name*]

Configuring a peer host public key

About peer host public key configuration

To encrypt information sent to a peer device or authenticate the digital signature of the peer device, you must configure the peer device's public key on the local device.

You can configure the peer host public key by using the following methods:

- Import the peer host public key from a public key file (recommended).
- Manually enter (type or copy) the peer host public key.

For information about how to obtain the host public key of a device, see "Distributing a local host public key."

Restrictions and guidelines for peer host public key configuration

When you configure a peer host public key, follow these restrictions and guidelines:

- When you manually enter the peer host public key, make sure the entered key is in the correct format. To obtain the peer host public key in the correct format, use the **display public-key local public** command to display the public key on the peer device and record the key. The format of the public key displayed in any other way might be incorrect. If the key is not in the correct format, the system discards the key and displays an error message.
- Always import rather than enter the peer host public key if you are not sure whether the device supports the format of the recorded peer host public key.

Importing a peer host public key from a public key file

About this task

Before you perform this task, make sure you have exported the host public key to a file on the peer device and obtained the file from the peer device. For information about exporting a host public key, see "Exporting a host public key."

After you import the key, the system automatically converts the imported public key to a string in the Public Key Cryptography Standards (PKCS) format.

Procedure

1. Enter system view.

   system-view

2. Import a peer host public key from a public key file.

   public-key peer *keyname* **import sshkey** *filename*

   By default, no peer host public keys exist.

Entering a peer host public key

About this task

Before you perform this task, make sure you have displayed the key on the peer device and recorded the key. For information about displaying a host public key, see "Displaying a host public key."

Procedure

1. Enter system view.

   system-view

2. Specify a name for the peer host public key and enter public key view.

   public-key peer *keyname*

3. Type or copy the key.

   You can use spaces and carriage returns, but the system does not save them.

4. Exit public key view.

   peer-public-key end

   When you exit public key view, the system automatically saves the peer host public key.

Destroying a local key pair

About this task

To ensure security, destroy the local key pair and generate a new key pair in any of the following situations:

- The local key has leaked. An intrusion event might occur.
- The storage media of the device is replaced.
- The local certificate has expired. For more information about local certificates, see "Configuring PKI."

Procedure

1. Enter system view.

   system-view

2. Destroy a local key pair.

   public-key local destroy { **dsa** | **ecdsa** | **rsa** | **sm2** } [**name** *key-name*]

Display and maintenance commands for public keys

Execute **display** commands in any view.

Task	Command
Display local public keys.	**display public-key local** { **dsa** | **ecdsa** | **rsa** | **sm2** } **public** [**name** *key-name*]

Task	Command
Display peer host public keys.	**display public-key peer** [**brief** | **name** *publickey-name*]

Examples of public key management

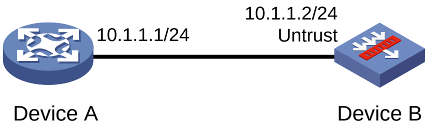
Example: Entering a peer host public key

Network configuration

As shown in Figure 2, to prevent illegal access, Device B authenticates Device A through a digital signature. Before configuring authentication parameters on Device B, use the following procedure to configure the public key of Device A on Device B:

- Create RSA key pairs on Device A and export the RSA host public key to a file.
- Manually specify the RSA host public key of Device A on Device B.

Figure 2 Network diagram



Procedure

1. Configure Device A:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Create local RSA key pairs with default names on Device A, and use the default modulus length 1024 bits.

```
[DeviceA] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
```

   # Display all local RSA public keys.

```
[DeviceA] display public-key local rsa public

=============================================
Key name: hostkey(default)
Key type: RSA
Time when key pair created: 10:28:00 2017/05/19
Key code:
   30819F300D06092A864886F70D010101050003818D0030818902818100C305DA3BD5EDB06D
   9828084DA4E4EA8E2FBA246BF29D85F71ABB6044CC5D4EC71E9A14220C38FE9EF2313E1FF6
   2A927E44CF24E1341AF76CEF0D65A7E8E2E43596E9B8EB568DD5CFCDE4CDC892FDED1BC861
   91E561D56E5E77C30D427D0D9D4B22E71572F725F63266C2D8E45BE87A7DAFE94204DAB702
   4A6E137BE307C5ACB70203010001

=============================================
Key name: serverkey(default)
Key type: RSA
Time when key pair created: 10:28:00 2017/05/19
Key code:

   307C300D06092A864886F70D0101010500036B003068026100CBF917CB6C6CFD28B92BC07E
   621326E3079464B31BB18D3D3AC866D48812CE14DBFC9EFC347A7E96D3714B7CF918A1751C
   681E5A3DBE79961E26829009308CAA3E920824BFE5B8907CCA073405CEDC5681B6AF82B7F9
   593FF85DF70D607B13430203010001
```

2. Configure Device B:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Add the interface to the security zone.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

   # Configure a rule named **pkeylocalin** to permit the packets from Device A to Device B.

```
[DeviceB-security-policy-ip] rule name pkeylocalin
[DeviceB-security-policy-ip-1-pkeylocalin] source-zone untrust
[DeviceB-security-policy-ip-1-pkeylocalin] destination-zone local
[DeviceB-security-policy-ip-1-pkeylocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-pkeylocalin] destination-ip-host 10.2.2.2
[DeviceB-security-policy-ip-1-pkeylocalin] action pass
[DeviceB-security-policy-ip-1-pkeylocalin] quit
[DeviceB-security-policy-ip] quit
```

   # Enter the host public key of Device A in public key view. The key must be literally the same as displayed on Device A.

```
<DeviceB> system-view
[DeviceB] public-key peer devicea
Enter public key view. Return to system view with "peer-public-key end" command.
[DeviceB-pkey-public-key-devicea]
30819F300D06092A864886F70D010101050003818D003081890
2818100C305DA3BD5EDB06D
[DeviceB-pkey-public-key-devicea]
9828084DA4E4EA8E2FBA246BF29D85F71ABB6044CC5D4EC7
1E9A14220C38FE9EF2313E1FF6
[DeviceB-pkey-public-key-devicea]
2A927E44CF24E1341AF76CEF0D65A7E8E2E43596E9B8EB56
```

```
8DD5CFCDE4CDC892FDED1BC861
[DeviceB-pkey-public-key-devicea]
91E561D56E5E77C30D427D0D9D4B22E71572F725F63266C2
D8E45BE87A7DAFE94204DAB702
[DeviceB-pkey-public-key-devicea] 4A6E137BE307C5ACB70203010001
```

# Save the public key and return to system view.

```
[DeviceB-pkey-public-key-devicea] peer-public-key end
```

Verifying the configuration

# Verify that the peer host public key configured on Device B is the same as the key displayed on Device A.

```
[DeviceB] display public-key peer name devicea

=============================================
Key name: devicea
Key type: RSA
Key modulus: 1024
Key code:

   30819F300D06092A864886F70D010101050003818D0030818902818100C305DA3BD5EDB06D
   9828084DA4E4EA8E2FBA246BF29D85F71ABB6044CC5D4EC71E9A14220C38FE9EF2313E1FF6
   2A927E44CF24E1341AF76CEF0D65A7E8E2E43596E9B8EB568DD5CFCDE4CDC892FDED1BC861
   91E561D56E5E77C30D427D0D9D4B22E71572F725F63266C2D8E45BE87A7DAFE94204DAB702
   4A6E137BE307C5ACB70203010001
```

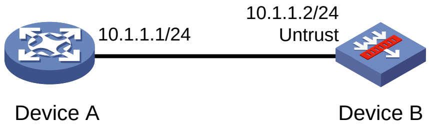
Example: Importing a public key from a public key file

Network configuration

As shown in Figure 3, Device B authenticates Device A through a digital signature. Before configuring authentication parameters on Device B, use the following procedure to configure the public key of Device A on Device B:

- Create RSA key pairs on Device A and export the RSA host public key to a file.
- Import the RSA host public key of Device A from the public key file to Device B.

Figure 3 Network diagram



Procedure

1. Configure Device A:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

2. Configure Device B:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Add the interface to the security zone.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

# Configure a rule named **pkeylocalin** to permit the packets from Device A to Device B.

```
[DeviceB-security-policy-ip] rule name pkeylocalin
[DeviceB-security-policy-ip-1-pkeylocalin] source-zone untrust
[DeviceB-security-policy-ip-1-pkeylocalin] destination-zone local
[DeviceB-security-policy-ip-1-pkeylocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-pkeylocalin] destination-ip-host 10.2.2.2
[DeviceB-security-policy-ip-1-pkeylocalin] action pass
[DeviceB-security-policy-ip-1-pkeylocalin] quit
[DeviceB-security-policy-ip] quit
```

3. Configure Device A:

# Create local RSA key pairs with default names on Device A, and use the default modulus length (1024 bits).

```
<DeviceA> system-view
[DeviceA] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.
Create the key pair successfully.
```

# Display all local RSA public keys.

```
[DeviceA] display public-key local rsa public

=============================================
Key name: hostkey(default)
Key type: RSA
Time when key pair created: 10:58:47 2017/05/19
Key code:

   30819F300D06092A864886F70D010101050003818D0030818902818100D584D31764E92A97
   11B6AE429BDF8A3C70D019D79D6F9905789E1020E5904121FB76261C48227451FE14B754EE
   1C0CBFCD96021B55D4E439F1377DBBE466323F7559ED7BC3BC94F7CA16362A9E0FB010567A
   F202B45906B71F90F75A09D72BA16EFB3F44D7B3AE90D450C4C72F478F57935FE4E8F6BFE1
   C2F404E7E8C13174790203010001

=============================================
Key name: serverkey(default)
Key type: RSA
Time when key pair created: 10:58:47 2017/05/19
```

```
Key code:

   307C300D06092A864886F70D0101010500036B003068026100BD8BED0D9EC4C8435B074FF4
   0EDE756A7DE9276638184DF53A7AB072EC75025C22F65B68C16B04278E158078B44445E8F7
   73BD4182616EB5F23089D214FE5A5E246B4C54C2B23491A690AA69C30CCE5E8705D8BFFDED
   C90292074A836A86BB8D0203010001
```

# Export the RSA host public key to file **devicea.pub**.

```
[DeviceA] public-key local export rsa ssh2 devicea.pub
```

# Enable the FTP server function, create an FTP user with username **ftp** and password **123**, and configure the FTP user role as **network-admin**.

```
[DeviceA] ftp server enable
[DeviceA] local-user ftp
[DeviceA-luser-manage-ftp] password simple 123
[DeviceA-luser-manage-ftp] service-type ftp
[DeviceA-luser-manage-ftp] authorization-attribute user-role network-admin
[DeviceA-luser-manage-ftp] quit
```

4. Configure Device B:

# Use FTP in binary mode to get public key file **devicea.pub** from Device A.

```
<DeviceB> ftp 10.1.1.1
Connected to 10.1.1.1 (10.1.1.1).
220 FTP service ready.
User (10.1.1.1:(none)): ftp
331 Password required for ftp.
Password:
230 User logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> binary
200 TYPE is now 8-bit binary
ftp> get devicea.pub
227 Entering Passive Mode (10,1,1,1,118,252)
150 Accepted data connection
226 File successfully transferred
301 bytes received in 0.022 seconds (13.45 Kbytes/s)
ftp> quit
221-Goodbye. You uploaded 0 and downloaded 1 kbytes.
221 Logout.
```

# Import the host public key from key file **devicea.pub**.

```
<DeviceB> system-view
[DeviceB] public-key peer devicea import sshkey devicea.pub
```

Verifying the configuration

# Verify that the peer host public key configured on Device B is the same as the key displayed on Device A.

```
[DeviceB] display public-key peer name devicea

=============================================
Key name: devicea
Key type: RSA
Key modulus: 1024
Key code:

   30819F300D06092A864886F70D010101050003818D0030818902818100D584D31764E92A97
   11B6AE429BDF8A3C70D019D79D6F9905789E1020E5904121FB76261C48227451FE14B754EE
   1C0CBFCD96021B55D4E439F1377DBBE466323F7559ED7BC3BC94F7CA16362A9E0FB010567A
   F202B45906B71F90F75A09D72BA16EFB3F44D7B3AE90D450C4C72F478F57935FE4E8F6BFE1
   C2F404E7E8C13174790203010001
```

Contents

Configuring PKI

About PKI

Public Key Infrastructure (PKI) is an asymmetric key infrastructure to encrypt and decrypt data for securing network services.

PKI uses digital certificates to distribute and employ public keys, and provides network communication and e-commerce with security services such as user authentication, data confidentiality, and data integrity. For more information about public keys, see "Managing public keys."

PKI terminology

Digital certificate

A digital certificate is an electronic document signed by a CA that binds a public key with the identity of its owner.

A digital certificate includes the following information:

- Issuer name (name of the CA that issued the certificate).
- Subject name (name of the individual or group to which the certificate is issued).
- Identity information of the subject.
- Subject's public key.
- Signature of the CA.
- Validity period.

A digital certificate must comply with the international standards of ITU-T X.509, of which X.509 v3 is the most commonly used.

This chapter covers the following types of certificates:

- **CA certificate**—Certificate of a CA. Multiple CAs in a PKI system form a CA tree, with the root CA at the top. The root CA generates a self-signed certificate, and each lower level CA holds a CA certificate issued by the CA immediately above it. The chain of these certificates forms a chain of trust.
- **Registration authority (RA) certificate**—Certificate issued by a CA to an RA. RAs act as proxies for CAs to process enrollment requests in a PKI system.
- **Local certificate**—Certificate of a local PKI entity, which contains the entity's public key. Local certificates include the following types:
  - **CA-signed certificate**—Digital certificate signed and issued by a CA to the local PKI entity.
  - **Self-signed certificate**—Digital certificate signed and issued by the local PKI entity (the device) itself. When the device cannot request a local certificate from the CA, it can sign and issue a local certificate for communication with the peer to provide services. Using this type of certificate is simple in configuration but has low security. You can use self-signed certificates in scenarios with low security requirements.
- **Peer certificate**—CA-signed digital certificate of a peer, which contains the peer's public key.

Fingerprint of root CA certificate

Each root CA certificate has a unique fingerprint, which is the hash value of the certificate content. The fingerprint of a root CA certificate can be used to authenticate the validity of the root CA.

Certificate revocation list

A certificate revocation list (CRL) is a list of serial numbers for certificates that have been revoked. A CRL is created and signed by the CA that originally issued the certificates.

The CA publishes CRLs periodically to revoke certificates. Entities that are associated with the revoked certificates should not be trusted.

The CA must revoke a certificate when any of the following conditions occurs:

- The certificate subject name is changed.
- The private key is compromised.
- The association between the subject and CA is changed. For example, when an employee terminates employment with an organization.

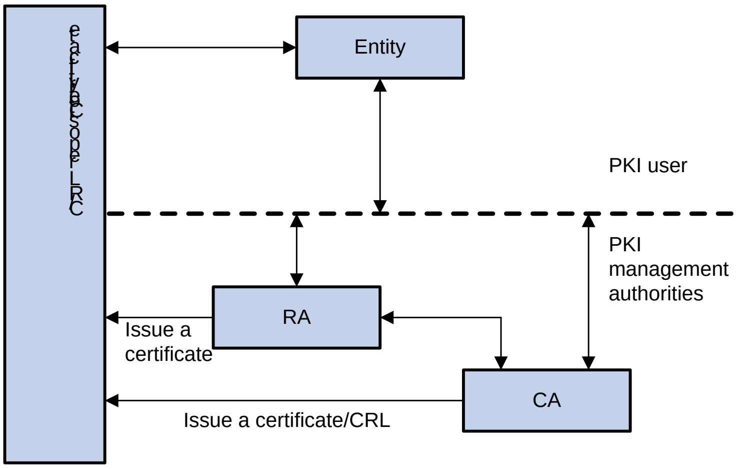
CA policy

A CA policy is a set of criteria that a CA follows to process certificate requests, to issue and revoke certificates, and to publish CRLs. Typically, a CA advertises its policy in a certification practice statement (CPS). You can obtain a CA policy through out-of-band means such as phone, disk, and email. Make sure you understand the CA policy before you select a trusted CA for certificate request because different CAs might use different policies.

PKI architecture

A PKI system consists of PKI entities, CAs, RAs and a certificate/CRL repository, as shown in Figure 1.

Figure 1 PKI architecture



PKI entity

A PKI entity is an end user using PKI certificates. A PKI entity can be an operator, an organization, a device like a router or a switch, or a process running on a computer. PKI entities use SCEP to communicate with the CA or RA.

CA

A certification authority (CA) grants and manages certificates. It issues certificates, defines the certificate validity periods, and revokes certificates by publishing CRLs.

RA

A registration authority (RA) offloads the CA by processing certificate enrollment requests. The RA accepts certificate requests, verifies user identity, and determines whether to ask the CA to issue certificates.

The RA is optional in a PKI system. In cases when there is security concern over exposing the CA to direct network access, it is advisable to delegate some of the tasks to an RA. Then, the CA can concentrate on its primary tasks of signing certificates and CRLs.

Certificate/CRL repository

A certificate/CRL repository is a certificate distribution point that stores certificates and CRLs, and distributes these certificates and CRLs to PKI entities. It also provides the query function. A PKI repository can be a directory server using the LDAP or HTTP protocol, of which LDAP is commonly used.

Retrieval, usage, and maintenance of a digital certificate

The following workflow describes the retrieval, usage, and maintenance of a digital certificate. This example uses a CA which has an RA to process certificate enrollment requests.

1. A PKI entity generates an asymmetric key pair and submits a certificate request to the RA.

   The certificate request contains the public key and its identity information.
2. The RA verifies the identity of the entity and sends a digital signature containing the identity information and the public key to the CA.
3. The CA verifies the digital signature, approves the request, and issues a certificate.
4. After receiving the certificate from the CA, the RA sends the certificate to the certificate repository and notifies the PKI entity that the certificate has been issued.
5. The PKI entity obtains the certificate from the certificate repository.
6. To establish a secure connection for communication, two PKI entities exchange local certificates to authenticate each other. The connection can be established only if both entities verify that the peer's certificate is valid.
7. You can remove the local certificate of a PKI entity and request a new one when any of the following conditions occur:
   - The local certificate is about to expire.
   - The certificate's private key is compromised.

PKI applications

The PKI technology can meet security requirements of online transactions. As an infrastructure, PKI has a wide range of applications. The PKI system of NSFOCUS can provide certificate management for IPsec and SSL.

The following are some application examples.

VPN

A VPN is a private data communication network built on the public communication infrastructure. A VPN can use network layer security protocols (for example, IPsec) in conjunction with PKI-based encryption and digital signature technologies for confidentiality.

Secure emails

PKI can address the email requirements for confidentiality, integrity, authentication, and non-repudiation. A common secure email protocol is Secure/Multipurpose Internet Mail Extensions (S/MIME), which is based on PKI and allows for transfer of encrypted mails with signature.

Web security

PKI can be used in the SSL handshake phase to verify the identities of the communicating parties by digital certificates.

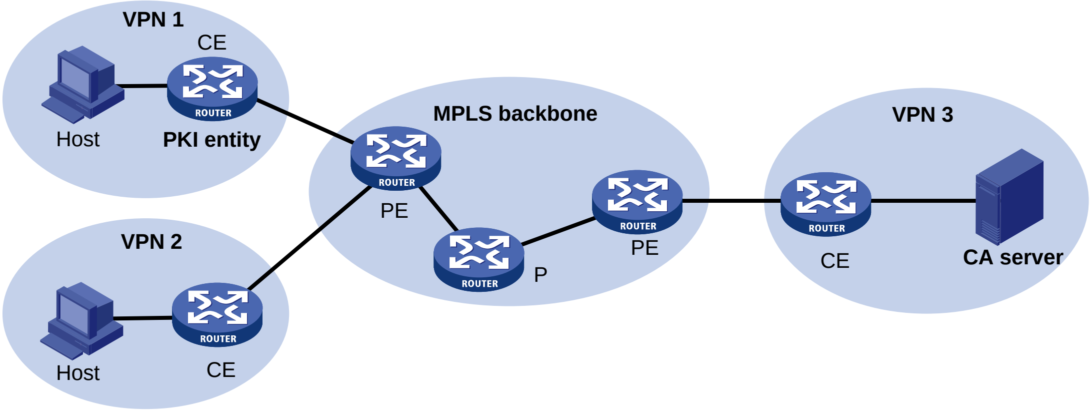
Support for VPN

An enterprise might have multiple branches in different VPNs. PKI support for VPN is required if users in different VPNs request certificates from the CA server in the headquarters VPN.

As shown in Figure 2, the PKI entity in VPN 1 requests a certificate from the CA server in VPN 3 in the following workflow:

1. The PKI entity submits a certificate request to the CA server.
2. The PE device connected to the PKI entity transmits the request to the CA server through the VPN instances.
3. The CA server verifies the request and issues the certificate.
4. The PE device connected to the CA server transmits the certificate to the PKI entity through the VPN instances.

Figure 2 PKI support for VPN



PKI tasks at a glance

To configure PKI, perform the following tasks:

1. Configuring a PKI entity
2. Configuring a PKI domain
3. (Optional.) Specifying the storage path for certificates and CRLs
4. Requesting a certificate

   Choose one of the following tasks:
   - Enabling the automatic online certificate request mode
   - Manually submitting an online certificate request
   - Manually submitting a certificate request in offline mode
5. (Optional.) Aborting a certificate request
6. (Optional.) Obtaining certificates

   You can obtain the CA certificate, local certificates, and peer certificates related to a PKI domain from a CA and save them locally for higher lookup efficiency.
7. (Optional.) Verifying PKI certificates
8. (Optional.) Exporting certificates
9. (Optional.) Removing a certificate
10. (Optional.) Configuring a certificate-based access control policy

Certificate-based access control policies allow you to authorize access to a device (for example, an HTTPS server) based on the attributes of an authenticated client's certificate.

11. (Optional.) Enabling local certificate expiration notification
12. (Optional.) Obtaining the CRL

Configuring a PKI entity

About PKI entities

A certificate applicant uses an entity to provide its identity information to a CA. A valid PKI entity must include one or more of following identity categories:

- Distinguished name (DN) of the entity, which further includes the common name, country code, locality, organization, unit in the organization, and state. If you configure the DN for an entity, a common name is required.
- FQDN of the entity.
- IP address of the entity.

Restrictions and guidelines for PKI entity configuration

Follow these restrictions and guidelines when you configure a PKI entity:

- Whether the identity categories are required or optional depends on the CA policy. Follow the CA policy to configure the entity settings. For example, if the CA policy requires the entity DN, but you configure only the IP address, the CA rejects the certificate request from the entity.
- The SCEP add-on on the Windows 2000 CA server has restrictions on the data length of a certificate request. If a request from a PKI entity exceeds the data length limit, the CA server does not respond to the certificate request. In this case, you can use an out-of-band means to submit the request. Other types of CA servers, such as RSA servers and OpenCA servers, do not have such restrictions.

PKI entity tasks at a glance

To configure a PKI entity, perform the following tasks:

1. Configuring the DN for the PKI entity
2. Configuring the FQDN for the PKI entity
3. Configuring the IP address for the PKI entity

Configuring the DN for the PKI entity

Restrictions and guidelines for configuring the DN for the PKI entity

You can configure the individual attributes used to construct the DN string for the PKI entity, or directly configure the full DN string by using the **subject-dn** command.

If the **subject-dn** command is configured, the individual DN attributes configured by using the **common-name**, **country**, **locality**, **organization, organization-unit**, and **state** commands do not take effect.

Configuring the individual DN attributes

1. Enter system view.

   ```
   system-view
   ```

2. Create a PKI entity and enter its view.

   pki entity *entity-name*

3. Set a common name for the entity.

   common-name *common-name-sting*

   By default, the common name is not set.

4. Set the country code of the entity.

   country *country-code-string*

   By default, the country code is not set.

5. Set the locality of the entity.

   locality *locality-name*

   By default, the locality is not set.

6. Set the organization of the entity.

   organization *org-name*

   By default, the organization is not set.

7. Set the unit of the entity in the organization.

   organization-unit *org-unit-name*

   By default, the unit is not set.

8. Set the state where the entity resides.

   state *state-name*

   By default, the state is not set.

Configuring the full DN string

1. Enter system view.

   system-view

2. Create a PKI entity and enter its view.

   pki entity *entity-name*

3. Configure the full subject DN string.

   subject-dn *dn-string*

   By default, the PKI entity DN is not configured.

Configuring the FQDN for the PKI entity

1. Enter system view.

   system-view

2. Create a PKI entity and enter its view.

   pki entity *entity-name*

3. Configure the FQDN for the PKI entity.

   fqdn *fqdn-name-string*

   By default, the FQDN is not configured.

Configuring the IP address for the PKI entity

1. Enter system view.

   system-view

2. Create a PKI entity and enter its view.

   pki entity *entity-name*

3. Configure the IP address for the PKI entity.

ip { *ip-address* | **interface** *interface-type interface-number* }

By default, the IP address is not configured.

Configuring a PKI domain

About PKI domains

A PKI domain contains enrollment information for a PKI entity. It is locally significant and is intended only for use by other applications like IKE and SSL.

PKI domain tasks at a glance

To configure a PKI domain, perform the following tasks:

1. Creating a PKI domain
2. Specifying the trusted CA
3. Specifying the PKI entity name
4. Specifying the certificate request reception authority
5. Specifying the certificate request URL
6. (Optional.) Specifying the VPN instance where the certificate request reception authority and the CRL repository belong
7. (Optional.) Setting the SCEP polling interval and maximum polling attempts
8. Specifying the LDAP server

   This task is required when either of the following conditions is met:

   - The device must obtain certificates from the CA by using the LDAP protocol.
   - An LDAP URL which does not contain the host name of the LDAP server is specified as the CRL repository URL.
9. Specifying the fingerprint for root CA certificate verification

   This step is required if the auto certificate request mode is configured in the PKI domain.

   If the manual certificate request mode is configured, you can skip this step and manually verify the fingerprint displayed during verification of the root CA certificate.
10. Specifying the key pair for certificate request
11. (Optional.) Specifying the intended purpose for the certificate
12. (Optional.) Specifying the source IP address for PKI protocol packets
13. (Optional.) Specifying the encryption algorithm for certificate files in PKCS#7 format

Creating a PKI domain

1. Enter system view.

   system-view
2. Create a PKI domain and enter its view.

   pki domain *domain-name*

Specifying the trusted CA

About this task

The PKI domain must have a CA certificate before you can request a local certificate. To obtain a CA certificate, the trusted CA name must be specified. The trusted CA name uniquely identifies the CA to be used if multiple CAs exist on the CA server.

Procedure

1. Enter system view.

   `system-view`
2. Enter PKI domain view.

   `pki domain` *`domain-name`*
3. Specify the trusted CA name.

   `ca identifier` *`name`*

   By default, no trusted CA name is specified.

Specifying the PKI entity name

1. Enter system view.

   `system-view`
2. Enter PKI domain view.

   `pki domain` *`domain-name`*
3. Specify the PKI entity name.

   `certificate request entity` *`entity-name`*

   By default, no PKI entity name is specified.

Specifying the certificate request reception authority

1. Enter system view.

   `system-view`
2. Enter PKI domain view.

   `pki domain` *`domain-name`*
3. Specify the certificate request reception authority.

   `certificate request from` { **`ca`** | **`ra`** }

   By default, no certificate request reception authority is specified.

Specifying the certificate request URL

1. Enter system view.

   `system-view`
2. Enter PKI domain view.

   `pki domain` *`domain-name`*
3. Specify the URL of the certificate request reception authority to which the device sends certificate requests.

   `certificate request url` *`url-string`*

   By default, the certificate request URL is not specified.

Specifying the VPN instance where the certificate request reception authority and the CRL repository belong

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Specify the VPN instance where the certificate request reception authority and the CRL repository belong.

   vpn-instance *vpn-instance-name*

   By default, the certificate request reception authority and the CRL repository belong to the public network.

Setting the SCEP polling interval and maximum polling attempts

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Set the SCEP polling interval and maximum number of polling attempts.

   certificate request polling { **count** *count* | **interval** *interval* }

   By default, the device polls the CA server for the certificate request status every 20 minutes. The maximum number of polling attempts is 50.

Specifying the LDAP server

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Specify the LDAP server.

   ldap-server host *hostname* [**port** *port-number*] [**vpn-instance** *vpn-instance-name*]

   By default, no LDAP server is specified.

Specifying the fingerprint for root CA certificate verification

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Configure the fingerprint for verifying the root CA certificate.

   root-certificate fingerprint { **md5** | **sha1** } *string*

   By default, no fingerprint is configured.

Specifying the key pair for certificate request

About this task

You can specify any of the following types of key pairs for certificate request in a PKI domain:

- DSA.
- ECDSA.
- RSA.
- SM2.

The private key of the key pair is kept secret. The public key of the key pair will be sent together with other information in a certificate request to the CA, which signs the request data and issues the certificate.

For more information about DSA, ECDSA, RSA, and SM2 key pairs, see "Managing public keys."

Restrictions and guidelines

You can specify a nonexistent key pair for certificate request. The PKI entity automatically creates the key pair before submitting a certificate request.

Procedure

1. Enter system view.
   system-view
2. Enter PKI domain view.
   pki domain *domain-name*
3. Specify the key pair for certificate request.
   - Specify an RSA key pair.
     public-key rsa { { **encryption name** *encryption-key-name* [**length** *key-length*] | **signature name** *signature-key-name* [**length** *key-length*] } * | **general name** *key-name* [**length** *key-length*] }
   - Specify an ECDSA key pair.
     public-key ecdsa name *key-name* [**secp192r1** | **secp256r1** | **secp384r1** | **secp521r1**]
   - Specify a DSA key pair.
     public-key dsa name *key-name* [**length** *key-length*]
   - Specify an SM2 key pair.
     public-key sm2 { { **encryption name** *encryption-key-name* | **signature name** *signature-key-name* } * | **general name** *key-name* }

   By default, no key pair is specified.

Specifying the intended purpose for the certificate

About this task

An issued certificate contains the extensions which restrict the usage of the certificate to specific purposes. You can specify the intended purposes for a certificate, which will be included in the certificate request sent to the CA. However, the actual extensions contained in an issued certificate depend on the CA policy, and they might be different from those specified in the PKI domain. Whether an application (such as IKE and SSL) will use the certificate during authentication depends on the application's policy.

Supported certificate extensions include:

- **ike**—Certificates carrying this extension can be used by IKE peers.
- **ssl-client**—Certificates carrying this extension can be used by SSL clients.
- **ssl-server**—Certificates carrying this extension can be used by SSL servers.

Procedure

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Specify the intended use for the certificate.

   usage { **ike** | **ssl-client** | **ssl-server** } *

   By default, the certificate can be used by all supported applications, including IKE, SSL client, and SSL server.

Specifying the source IP address for PKI protocol packets

About this task

This task is required if the CA policy requires that the CA server accept certificate requests from a specific IP address or subnet.

Procedure

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Specify a source IP address for the PKI protocol packets.

   IPv4:

   source ip { *ip-address* | **interface** *interface-type interface-number* }

   IPv6:

   source ipv6 { *ipv6-address* | **interface** *interface-type interface-number* }

   By default, the source IP address of PKI protocol packets is the IP address of their outgoing interface.

Specifying the encryption algorithm for certificate files in PKCS#7 format

About this task

During online certificate request, the device uses the specified encryption algorithm to encrypt the certificate signing request in PKCS#7 format before sending the request to the CA. After obtaining the certificate issued by the CA, the device uses the encryption algorithm to decrypt the certificate file in PKCS#7 format. Make sure the specified encryption algorithm is supported on the CA server.

Procedure

1. Enter system view.

   system-view

2. Enter PKI domain view.

pki domain *domain-name*

3. Specify the encryption algorithm for certificate files in PKCS#7 format.

   pkcs7-encryption-algorithm { **3des-cbc** | **aes-cbc-128** | **des-cbc** | **sm4-cbc** }

   By default, the DES-CBC encryption algorithm is used.

Specifying the storage path for certificates and CRLs

About this task

The device has a default storage path for certificates and CRLs. You can change the storage path and specify different paths for the certificates and CRLs.

After you change the storage path for certificates or CRLs, the certificate files and CRL files in the original path are moved to the new path. Certificate files use the .cer or .p12 file extension and CRL files use the .crl file extension.

Restrictions and guidelines

If you change the storage path, save the configuration before you reboot or shut down the device to avoid loss of certificates or CRLs.

Procedure

1. Enter system view.

   system-view

2. Specify the storage path for certificates and CRLs.

   pki storage { **certificates** | **crls** } *dir-path*

   By default, the device stores certificates and CRLs in the PKI directory on the storage media of the device.

Requesting a certificate

About certificate request configuration

To request a certificate, a PKI entity must provide its identity information and public key to a CA.

A certificate request can be submitted to a CA in offline or online mode.

- **Offline mode**—A certificate request is submitted by using an out-of-band method, such as phone, disk, or email.
- **Online mode**—A certificate request can be automatically or manually submitted to a CA through the Simple Certificate Enrollment Protocol (SCEP).

Restrictions and guidelines for certificate request configuration

When you request a local certificate in a PKI domain, follow these restrictions and guidelines:

- To prevent an existing local certificate from becoming invalid, do not perform the following tasks:
  - Create a key pair with the same name as the key pair contained in the certificate.

To create a key pair, use the **`public-key local create`** command.

  - Destroy the key pair contained in the certificate.

    To destroy a key pair, use the **`public-key local destroy`** command.

  For more information about the **`public-key local create`** and **`public-key local destroy`** commands, see public key management commands in *Security Command Reference*.

- To manually request a new certificate in a PKI domain that already has a local certificate, use the following procedure:
  - **a.** Use the **`pki delete-certificate`** command to delete the existing local certificate.
  - **b.** Use the **`public-key local create`** command to generate a new key pair.
  - **c.** Manually submit a certificate request.
- A PKI domain can have local certificates using only one type of cryptographic algorithms (DSA, ECDSA, RSA, or SM2). If DSA or ECDSA is used, a PKI domain can have only one local certificate. If RSA or SM2 is used, a PKI domain can have one local certificate for signature, and one local certificate for encryption.

Prerequisites for certificate request configuration

Make sure the device is time synchronized with the CA server. If the device is not time synchronized with the CA server, the certificate request might fail because the certificate might be considered to be outside of the validity period. For information about configuring the system time, see device management in *Fundamentals Configuration Guide*.

Enabling the automatic online certificate request mode

About this task

In auto request mode, a PKI entity with no local certificates automatically submits a certificate request to the CA when an application works with the PKI entity. For example, when IKE negotiation uses a digital signature for identity authentication, but no local certificate is available, the entity automatically submits a certificate request. It saves the certificate locally after obtaining the certificate from the CA.

A CA certificate must be present before you request a local certificate. If no CA certificate exists in the PKI domain, the PKI entity automatically obtains a CA certificate before sending a certificate request.

Certificate auto-renewal enables the system to automatically request a new certificate the specified number of days before the old certificate expires. The old certificate is replaced immediately when the new certificate is received.

Restrictions and guidelines

In auto request mode, the device does not automatically request a new certificate if the current certificate is about to expire or has expired, which might cause service interruptions.

To avoid service interruptions caused by certificate expiration, specify the **`renew-before-expire`** *days* option to enable certificate auto-renewal in auto certificate request mode.

Some CAs require a new PKI entity common name for certificate auto-renewal to work. Specify the **`automatic-append common-name`** keyword to ensure successful certificate auto-renewal.

Procedure

1. Enter system view.

```
system-view
```

2. Enter PKI domain view.

   pki domain *domain-name*

3. Enable the automatic online certificate request mode.

   certificate request mode auto [**password** { **cipher** | **simple** } *string* | **renew-before-expire** *days* [**reuse-public-key**] [**automatic-append common-name**]] *

   By default, the manual request mode applies.

   If the CA policy requires a password for certificate revocation, specify the password in this command.

Manually submitting an online certificate request

About this task

In manual request mode, you must execute the **pki request-certificate domain** command to request a local certificate in a PKI domain. The certificate will be saved in the domain after it is obtained from the CA.

Procedure

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Set the certificate request mode to manual.

   certificate request mode manual

   By default, the manual request mode applies.

4. Return to system view.

   quit

5. Obtain a CA certificate.

   See "Obtaining certificates."

   This step is required if the PKI domain does not have a CA certificate. The CA certificate is used to verify the authenticity and validity of the obtained local certificate.

6. Manually submit an SCEP certificate request.

   pki request-certificate domain *domain-name* [**password** *password*]

   This command is not saved in the configuration file.

   If the CA policy requires a password for certificate revocation, specify the password in this command.

Manually submitting a certificate request in offline mode

About this task

Use this method if the CA does not support SCEP or if a network connection between the device and CA is not possible.

Procedure

1. Enter system view.

   system-view

2. Enter PKI domain view.

pki domain *domain-name*

3. Set the certificate request mode to manual.

   certificate request mode manual

   By default, the manual request mode applies.

4. Return to system view.

   quit

5. Obtain the CA certificate.

   See "Obtaining certificates."

   This step is required if the PKI domain does not have a CA certificate. The CA certificate is used to verify the authenticity and validity of the obtained local certificate.

6. Print the certificate request in PKCS10 format on the terminal or save the certificate request to a PKCS10 file.

   pki request-certificate domain *domain-name* **pkcs10** [**filename** *filename*]

   This command is not saved in the configuration file.

7. Transfer certificate request information to the CA by using an out-of-band method.

8. Transfer the issued local certificate from the CA to the local device by using an out-of-band method.

9. Import the local certificate to the PKI domain.

   pki import domain *domain-name* { **der local filename** *filename* | **p12 local filename** *filename* | **pem local** [**filename** *filename*] }

Aborting a certificate request

About this task

Before the CA issues a certificate, you can abort a certificate request and change its parameters, such as the common name, country code, or FQDN. You can use the **display pki certificate request-status** command to display the status of a certificate request.

Alternatively, you also can remove a PKI domain to abort the associated certificate request.

Procedure

1. Enter system view.

   system-view

2. Abort a certificate request.

   pki abort-certificate-request domain *domain-name*

   This command is not saved in the configuration file.

Obtaining certificates

About this task

You can obtain the CA certificate, local certificates, and peer certificates related to a PKI domain from a CA and save them locally for higher lookup efficiency. To do so, use either the offline mode or the online mode:

- In offline mode, obtain the certificates by an out-of-band means like FTP, disk, or email, and then import them locally. Use this mode when the CRL repository is not specified, the CA server does not support SCEP, or the CA server generates the key pair for the certificates.

- In online mode, you can obtain the CA certificate through SCEP and obtain local certificates or peer certificates through LDAP.

Restrictions and guidelines

Follow these restrictions and guidelines when obtain certificates from a CA

- If a CA certificate already exists locally, you cannot obtain it again in online mode. If you want to obtain a new CA certificate, use the **`pki delete-certificate`** command to delete the existing CA certificate and local certificates first.
- If local or peer certificates already exist, you can obtain new local or peer certificates to overwrite the existing ones. If RSA is used, a PKI domain can have two local certificates, one for signature and the other for encryption.
- If CRL checking is enabled, obtaining a certificate triggers CRL checking. If the certificate to be obtained has been revoked, the certificate cannot be obtained.
- The device compares the validity period of a certificate with the local system time to determine whether the certificate is valid. Make sure the system time of the device is synchronized with the CA server.

Prerequisites

- Before you obtain local or peer certificates in online mode, make sure an LDAP server is correctly configured in the PKI domain.
- Before you import certificates in offline mode, complete the following tasks:
  - Use FTP or TFTP to upload the certificate files to the storage media of the device.

    If FTP or TFTP is not available, display and copy the contents of a certificate to a file on the device. Make sure the certificate is in PEM format because only certificates in PEM format can be imported.
  - Before you import a local certificate or peer certificate, obtain the CA certificate chain that signs the certificate.

    This step is required only if the CA certificate chain is neither available in the PKI domain nor contained in the certificate to be imported.
  - Before you import a local certificate that contains an encrypted key pair, contact the CA administrator to obtain the password required for importing the certificate.

Procedure

1. Enter system view.

   `system-view`
2. Obtain certificates.
   - Import certificates in offline mode.

     `pki import domain` *`domain-name`* { **`der`** { **`ca`** | **`local`** | **`peer`** } **`filename`** *`filename`* | **`p12 local filename`** *`filename`* | **`pem`** { **`ca`** | **`local`** | **`peer`** } [**`filename`** *`filename`*] }
   - Obtain certificates in online mode.

     `pki retrieve-certificate domain` *`domain-name`* { **`ca`** | **`local`** | **`peer`** *`entity-name`* }

     This command is not saved in the configuration file.

Verifying PKI certificates

About certification verification

A certificate is automatically verified when it is requested, obtained, or used by an application. If the certificate expires, if it is not issued by a trusted CA, or if it is revoked, the certificate cannot be used You can also manually verify a certificate.

You can enable or disable CRL checking in a PKI domain. CRL checking checks whether a certificate is in the CRL. If it is, the certificate has been revoked and its home entity is not trusted.

To use CRL checking, a CRL must be obtained from a CRL repository. The device selects a CRL repository in the following order:

1. CRL repository specified in the PKI domain by using the **`crl url`** command.
2. CRL repository in the certificate that is being verified.
3. CRL repository in the CA certificate or CRL repository in the upper-level CA certificate if the certificate being verified is a CA certificate

If no CRL repository is found after the selection process, the device obtains the CRL through SCEP. In this scenario, the CA certificate and the local certificates must have been obtained.

A certificate fails CRL checking in the following situations:

- A CRL cannot be obtained during CRL checking of the certificate.
- CRL checking verifies that the certificate has been revoked.

Restrictions and guidelines for certificate verification

When verifying the CA certificate of a PKI domain, the system needs to verify all the certificates in the CA certificate chain. To ensure a successful certificate verification process, the device must have all the PKI domains to which the CA certificates in the certificate chain belong.

The system verifies the CA certificates in the CA certificate chain as follows:

1. Identifies the parent certificate of the lowest-level certificate.

   Each CA certificate contains an issuer field that identifies the parent CA that issued the certificate.
2. Locates the PKI domain to which the parent certificate belongs.
3. Performs CRL checking in the PKI domain to check whether the parent certificate has been revoked. If it has been revoked, the certificate cannot be used.

   This step will not be performed when CRL checking is disabled in the PKI domain.
4. Repeats the previous steps for upper-level certificates in the CA certificate chain until the root CA certificate is reached.
5. Verifies that each CA certificate in the certificate chain is issued by the named parent CA, starting from the root CA.

Specifying the certificate revocation checking methods

About this task

You can specify the following certificate revocation checking methods in a PKI domain:

- **CRL**—Uses the CRL for certificate revocation checking.
- **None**—Performs no revocation checking and treats all certificates as not revoked.

The method specified first will be used first. If you specify **CRL** as method 1 and **none** as method 2, the device will first check the CRL to identify whether the certificate has been revoked during certificate verification. If the device cannot obtain a CRL, it will treat the certificate as not revoked.

If you specify **none** as method 1, you cannot specify method 2 because no revocation checking is required.

Restrictions and guidelines

The CRL method takes effect only if CRL checking is enabled in the PKI domain (by using the **`crl check enable`** command). If CRL checking is disabled in the PKI domain, no CRL checking will be performed and all certificates will be treated as not revoked.

Procedure

1. Enter system view.

   `system-view`

2. Enter PKI domain view.

   `pki domain` *`domain-name`*

3. Specify the certificate revocation checking methods.
   - Specify both the **CRL** and **none** certificate revocation checking methods.

     `revocation-check method crl` [**`none`**]
   - Specify the **none** certificate revocation checking method.

     `revocation-check method none`

   By default, the CRL is used for certificate revocation checking.

Verifying certificates with CRL checking

Restrictions and guidelines

CRL checking does not take effect if the **`revocation-check method none`** command is configured in the PKI domain.

Procedure

1. Enter system view.

   `system-view`

2. Enter PKI domain view.

   `pki domain` *`domain-name`*

3. (Optional.) Specify the URL of the CRL repository.

   `crl url` *`url-string`*

   By default, the URL of the CRL repository is not specified.

4. (Optional.) Specify the VPN instance where the certificate request reception authority and the CRL repository belong.

   `vpn-instance` *`vpn-instance-name`*

   By default, the certificate request reception authority and the CRL repository belong to the public network.

5. Enable CRL checking.

   `crl check enable`

   By default, CRL checking is enabled.

6. Return to system view.

   `quit`

7. Obtain the CA certificate.

   See "Obtaining certificates."

   The PKI domain must have a CA certificate before you can verify certificates in it.

8. (Optional.) Obtain the CRL and save it locally.

   See "Obtaining the CRL."

   To verify a non-root CA certificate and local certificates, the device automatically retrieves the CRL if the PKI domain has no CRL.

   The newly obtained CRL overwrites the old one, if any.

   The obtained CRL is issued by a CA in the CA certificate chain stored in the PKI domain.

9. Manually verify the validity of the certificates.

   pki validate-certificate domain *domain-name* { **ca** | **local** }

Verifying certificates without CRL checking

1. Enter system view.

   system-view

2. Enter PKI domain view.

   pki domain *domain-name*

3. Disable CRL checking.

   undo crl check enable

   By default, CRL checking is enabled.

4. Return to system view.

   quit

5. Obtain a CA certificate for the PKI domain.

   See "Obtaining certificates."

   The PKI domain must have a CA certificate before you can verify certificates in it.

6. Manually verify the certificate validity.

   pki validate-certificate domain *domain-name* { **ca** | **local** }

   This command is not saved in the configuration file.

Exporting certificates

About this task

You can export the CA certificate and the local certificates in a PKI domain to certificate files. The exported certificate files can then be imported back to the device or other PKI applications.

Restrictions and guidelines

To export all certificates in PKCS12 format, the PKI domain must have a minimum of one local certificate. If the PKI domain does not have any local certificates, the certificates in the PKI domain cannot be exported.

If you do not specify a file name when you export a certificate in PEM format, this command displays the certificate content on the terminal.

When you export a local certificate with RSA key pairs to a file, the certificate file name might be different from the file name specified in the command. The actual certificate file name depends on the purpose of the key pair contained in the certificate. For more information about the file naming rule, see the **pki export** command in *Security Command Reference*.

Procedure

1. Enter system view.

   system-view

2. Export certificates.

   - Export certificates in DER format.

     pki export domain *domain-name* **der** { **all** | **ca** | **local** } **filename** *filename*

   - Export certificates in PKCS12 format.

     pki export domain *domain-name* **p12** { **all** | **local** } **passphrase** *p12-key* **filename** *filename*

   - Export certificates in PEM format.

     pki export domain *domain-name* **pem** { { **all** | **local** } [{ **3des-cbc** | **aes-128-cbc** | **aes-192-cbc** | **aes-256-cbc** | **des-cbc** } *pem-key*] | **ca** } [**filename** *filename*]

Removing a certificate

About this task

You can remove certificates from a PKI domain in the following situations:

- Remove a CA certificate, local certificate, or peer certificate if the certificate has expired or is about to expire.
- Remove a local certificate if the certificate's private key is compromised, or if you want to request a new local certificate to replace the existing one.

Restrictions and guidelines

After you remove the CA certificate, the system automatically removes the local certificates, peer certificates, and CRLs from the domain.

To remove a local certificate and request a new certificate, perform the following tasks:

1. Remove the local certificate.
2. Use the **public-key local destroy** command to destroy the existing local key pair.
3. Use the **public-key local create** command to generate a new key pair.
4. Request a new certificate.

For more information about the **public-key local destroy** and **public-key local create** commands, see *Security Command Reference*.

Procedure

1. Enter system view.

   system-view

2. Remove a certificate.

   pki delete-certificate domain *domain-name* { **ca** | **local** | **peer** [**serial** *serial-num*] }

   If you use the **peer** keyword without specifying a serial number, this command removes all peer certificates.

Configuring a certificate-based access control policy

About certificate-based access control policies

Certificate-based access control policies allow you to authorize access to a device (for example, an HTTPS server) based on the attributes of an authenticated client's certificate.

Access control rules and certificate attribute groups

A certificate-based access control policy is a set of access control rules (permit or deny statements), each associated with a certificate attribute group. A certificate attribute group contains multiple attribute rules, each defining a matching criterion for an attribute in the certificate issuer name, subject name, or alternative subject name field.

Certificate matching mechanism

If a certificate matches all attribute rules in a certificate attribute group associated with an access control rule, the system determines that the certificate matches the access control rule. In this scenario, the match process stops, and the system performs the access control action defined in the access control rule.

The following conditions describe how a certificate-based access control policy verifies the validity of a certificate:

- If a certificate matches a permit statement, the certificate passes the verification.
- If a certificate matches a deny statement or does not match any statements in the policy, the certificate is regarded invalid.
- If a statement is associated with a non-existing attribute group, or the attribute group does not have attribute rules, the certificate matches the statement.
- If the certificate-based access control policy specified for a security application (for example, HTTPS) does not exist, all certificates in the application pass the verification.

Procedure

1. Enter system view.

   system-view

2. Create a certificate attribute group and enter its view.

   pki certificate attribute-group *group-name*

3. Configure an attribute rule for issuer name, subject name, or alternative subject name.

   attribute *id* { **alt-subject-name** { **fqdn** | **ip** } | { **issuer-name** | **subject-name** } { **dn** | **fqdn** | **ip** } } { **ctn** | **equ** | **nctn** | **nequ**} *attribute-value*

   By default, not attribute rules are configured.

4. Return to system view.

   quit

5. Create a certificate-based access control policy and enter its view.

   pki certificate access-control-policy *policy-name*

   By default, no certificate-based access control policies exist.

6. Create a certificate access control rule.

   rule [*id*] { **deny** | **permit** } *group-name*

By default, no certificate access control rules are configured, and all certificates can pass the verification.

You can create multiple certificate access control rules for a certificate-based access control policy.

Enabling local certificate expiration notification

About this task

After this feature is enabled, the system checks the validity date for local certificates every other hour. When a local certificate is about to expire in 30 days (included) or has expired, the system sends a notification log message for the certificate every other day.

Procedure

1. Enter system view.

   system-view

2. Enable local certificate expiration notification.

   pki certificate logging { **local-will-expire** | **local-has-expired** } **enable**

   By default, local certificate expiration notification is disabled.

Obtaining the CRL

About the task

The device uses HTTP, LDAP, or SCEP to obtain the latest CRL from the CRL repository to update the local CRL, if any.

The CRL can be obtained manually or automatically.

- **Manually obtain the CRL**—Execute the **pki retrieve-crl** command on the device to immediately obtain the latest CRL from the CRL repository.
- **Automatically obtain the CRL**—Execute the **crl update-period** command to enable automatic CRL update and specify the update interval. The device automatically connects to the CRL repository to obtain the CRL at the specified update intervals. This method is applicable to scenarios that require strict certificate verification, such as bank systems.

Restrictions and guidelines

The obtained CRL must be issued by a CA in the CA certificate chain stored in the PKI domain.

Using automatic CRL update, the device might not be able to update the CRL immediately when the CRL expires. This is because the device must wait for the specified interval of time to perform a next update. Set the CRL update interval to a proper value to ensure the timeliness of CRL update.

Prerequisites

Before obtaining the CRL, complete the following tasks:

- Specify the trusted CA by using the **ca identifier** command.
- Make sure the PKI domain contains the CA certificate. To obtain a CA certificate, use the **pki retrieve-certificate domain** command.
- Synchronize the system time of the device with that of the CA server. Time inconsistency might result in failure of obtaining the CRL. To set the system time, see device management in *Fundamentals Configuration Guide*.

Obtaining the CRL manually

1. Enter system view.

   `system-view`

2. (Optional.) Execute the following commands in sequence to specify the URL of the CRL repository.

   a. Enter PKI domain view.

      `pki domain` *`domain-name`*

   b. Specify the URL of the CRL repository.

      `crl url` *`url-string`*

   c. Return to the system view.

      `quit`

   If you do not specify the URL of the CRL repository, make sure the CA certificate or the local certificates contain CRL repository information.

3. Obtain the CRL manually.

   `pki retrieve-crl domain` *`domain-name`*

Obtaining the CRL automatically

1. Enter system view.

   `system-view`

2. Enter PKI domain view.

   `pki domain` *`domain-name`*

3. (Optional.) Specify the URL of the CRL repository.

   `crl url` *`url-string`*

   If you do not specify the URL of the CRL repository, make sure the CA certificate or the local certificates contain CRL repository information.

4. Enable automatic CRL update and set the update interval.

   `crl update-period` *`hours`*

   By default, the device does not automatically update the CRL.

Display and maintenance commands for PKI

Execute **display** commands in any view.

Task	Command
Display certificate-based access control policy information.	**display pki certificate access-control-policy** [*policy-name*]
Display certificate attribute group information.	**display pki certificate attribute-group** [*group-name*]
Display the contents of a certificate.	**display pki certificate domain** *domain-name* { **ca** | **local** | **peer** [**serial** *serial-num*] }
Display the certificate renewal status.	**display pki certificate renew-status** [**domain** *domain-name*]
Display certificate request status.	**display pki certificate request-status** [**domain** *domain-name*]

Task	**Command**
Display locally stored CRLs in a PKI domain.	**display pki crl domain** *domain-name*

PKI configuration examples

General restrictions and guidelines

You can use different software applications, such as Windows server, RSA Keon, and OpenCA, to act as the CA server.

If you use Windows server or OpenCA, you must install the SCEP add-on for Windows server or enable SCEP for OpenCA. In either case, when you configure a PKI domain, you must use the **certificate request from ra** command to specify the RA to accept certificate requests.

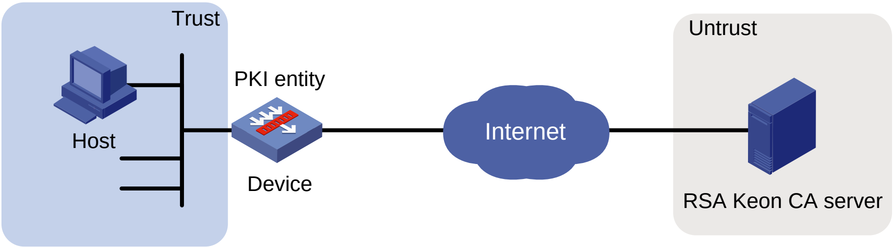
If you use RSA Keon, the SCEP add-on is not required. When you configure a PKI domain, you must use the **certificate request from ca** command to specify the CA to accept certificate requests.

Example: Requesting a certificate from an RSA Keon CA server

Network configuration

Configure the PKI entity (the device) to request a local certificate from the RSA Keon CA server.

Figure 3 Network diagram



Configuring the RSA Keon CA server

1. Create a CA server named **myca**:

   In this example, you must configure these basic attributes on the CA server:

   - **Nickname**—Name of the trusted CA.
   - **Subject DN**—DN attributes of the CA, including the common name (CN), organization unit (OU), organization (O), and country (C).

   You can use the default values for other attributes.

2. Configure extended attributes:

   Configure parameters in the **Jurisdiction Configuration** section on the management page of the CA server:

   - Select the correct extension profiles.
   - Enable the SCEP autovetting function to enable the CA server to automatically approve certificate requests without manual intervention.
   - Specify the IP address list for SCEP autovetting.

Configuring the device

1. Synchronize the system time of the device with the CA server for the device to correctly request certificates or obtain CRLs. (Details not shown.)
2. Assign IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

3. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 3.3.3.1 24 2.2.2.2
```

4. Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

5. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones:

# Configure a rule named **pkilocalout** to allow the device to send packets to the RSA Keon CA server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name pkilocalout
[Device-security-policy-ip-1-pkilocalout] source-zone local
[Device-security-policy-ip-1-pkilocalout] destination-zone untrust
[Device-security-policy-ip-1-pkilocalout] source-ip-host 2.2.2.1
[Device-security-policy-ip-1-pkilocalout] destination-ip-host 3.3.3.1
[Device-security-policy-ip-1-pkilocalout] action pass
[Device-security-policy-ip-1-pkilocalout] quit
```

# Configure a rule named **pkilocalin** to allow the device to receive the packets sent from the RSA Keon CA server.

```
[Device-security-policy-ip] rule name pkilocalin
[Device-security-policy-ip-2-pkilocalin] source-zone untrust
[Device-security-policy-ip-2-pkilocalin] destination-zone local
[Device-security-policy-ip-2-pkilocalin] source-ip-host 3.3.3.1
[Device-security-policy-ip-2-pkilocalin] destination-ip-host 2.2.2.1
[Device-security-policy-ip-2-pkilocalin] action pass
[Device-security-policy-ip-2-pkilocalin] quit
```

6. Create an entity named **aaa** and set the common name to **Device**.

```
[Device] pki entity aaa
[Device-pki-entity-aaa] common-name Device
[Device-pki-entity-aaa] quit
```

7. Configure a PKI domain and configure parameters for PKI certificate request.

# Create a PKI domain named **torsa** and enter its view.

```
[Device] pki domain torsa
```

# Specify the name of the trusted CA. The setting must be the same as CA name configured on the CA server. This example uses **myca**.

```
[Device-pki-domain-torsa] ca identifier myca
```

# Configure the URL of the CA server. The URL format is **http://***host:port***/***Issuing Jurisdiction ID*, where *Issuing Jurisdiction ID* is a hexadecimal string generated on the CA server.

```
[Device-pki-domain-torsa] certificate request url
http://1.1.2.22:446/80f6214aa8865301d07929ae481c7ceed99f95bd
```

# Configure the device to send certificate requests to **ca**.

```
[Device-pki-domain-torsa] certificate request from ca
```

# Set the PKI entity name to **aaa**.

```
[Device-pki-domain-torsa] certificate request entity aaa
```

# Specify the URL of the CRL repository.

```
[Device-pki-domain-torsa] crl url ldap://1.1.2.22:389/CN=myca
```

# Configure a general-purpose RSA key pair named **abc** with a length of 1024 bits.

```
[Device-pki-domain-torsa] public-key rsa general name abc length 1024
[Device-pki-domain-torsa] quit
```

8. Generate the RSA key pair.

```
[Device] public-key local create rsa name abc
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512,it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.........................++++++
....................................++++++
Create the key pair successfully.
```

9. Request a local certificate:

# Obtain the CA certificate and save it locally.

```
[Device] pki retrieve-certificate domain torsa ca
The trusted CA's finger print is:
    MD5  fingerprint:EDE9 0394 A273 B61A F1B3 0072 A0B1 F9AB
    SHA1 fingerprint: 77F9 A077 2FB8 088C 550B A33C 2410 D354 23B2 73A8
Is the finger print correct?(Y/N):y
Retrieved the certificates successfully.
```

# Submit a certificate request manually and set the certificate revocation password to **1111**. The certificate revocation password is required when an RSA Keon CA server is used.

```
[Device] pki request-certificate domain torsa password 1111
Start to request certificate ...
……
Request certificate of domain torsa successfully
```

Verifying the configuration

# Display information about the local certificate in PKI domain **torsa**.

```
[Device] display pki certificate domain torsa local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            15:79:75:ec:d2:33:af:5e:46:35:83:bc:bd:6e:e3:b8
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: CN=myca
        Validity
            Not Before: Jan  6 03:10:58 2013 GMT
            Not After : Jan  6 03:10:58 2014 GMT
        Subject: CN=Device
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:ab:45:64:a8:6c:10:70:3b:b9:46:34:8d:eb:1a:
                    a1:b3:64:b2:37:27:37:9d:15:bd:1a:69:1d:22:0f:
                    3a:5a:64:0c:8f:93:e5:f0:70:67:dc:cd:c1:6f:7a:
                    0c:b1:57:48:55:81:35:d7:36:d5:3c:37:1f:ce:16:
                    7e:f8:18:30:f6:6b:00:d6:50:48:23:5c:8c:05:30:
                    6f:35:04:37:1a:95:56:96:21:95:85:53:6f:f2:5a:
                    dc:f8:ec:42:4a:6d:5c:c8:43:08:bb:f1:f7:46:d5:
                    f1:9c:22:be:f3:1b:37:73:44:f5:2d:2c:5e:8f:40:
                    3e:36:36:0d:c8:33:90:f3:9b
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 CRL Distribution Points:

                Full Name:
                  DirName: CN = myca

    Signature Algorithm: sha1WithRSAEncryption
        b0:9d:d9:ac:a0:9b:83:99:bf:9d:0a:ca:12:99:58:60:d8:aa:
        73:54:61:4b:a2:4c:09:bb:9f:f9:70:c7:f8:81:82:f5:6c:af:
        25:64:a5:99:d1:f6:ec:4f:22:e8:6a:96:58:6c:c9:47:46:8c:
        f1:ba:89:b8:af:fa:63:c6:c9:77:10:45:0d:8f:a6:7f:b9:e8:
        25:90:4a:8e:c6:cc:b8:1a:f8:e0:bc:17:e0:6a:11:ae:e7:36:
        87:c4:b0:49:83:1c:79:ce:e2:a3:4b:15:40:dd:fe:e0:35:52:
        ed:6d:83:31:2c:c2:de:7c:e0:a7:92:61:bc:03:ab:40:bd:69:
        1b:f5
```

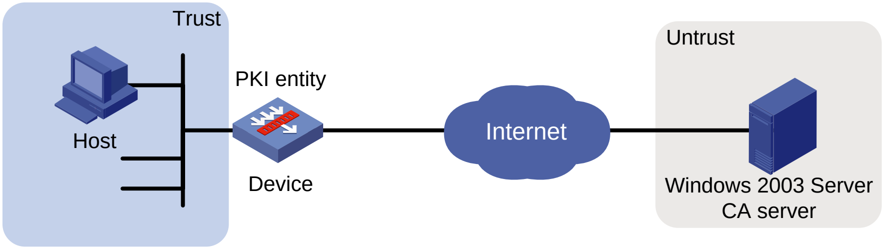
To display detailed information about the CA certificate, use the **display pki certificate domain** command.

Example: Requesting a certificate from a Windows Server 2003 CA server

Network configuration

Configure the PKI entity (the device) to request a local certificate from a Windows Server 2003 CA server.

Figure 4 Network diagram



Configuring the Windows Server 2003 CA server

1. Install the certificate service component:
   a. Select **Control Panel** > **Add or Remove Programs** from the start menu.
   b. Select **Add/Remove Windows Components** > **Certificate Services**.
   c. Click **Next** to begin the installation.
   d. Set the CA name. In this example, set the CA name to **myca**.
2. Install the SCEP add-on:

   By default, Windows Server 2003 does not support SCEP. You must install the SCEP add-on on the server for a PKI entity to register and obtain a certificate from the server. After the SCEP add-on installation is complete, you will see a URL. Specify this URL as the certificate request URL on the device.
3. Modify the certificate service attributes:
   a. Select **Control Panel** > **Administrative Tools** > **Certificate Authority** from the start menu.

      If the certificate service component and SCEP add-on have been installed successfully, there should be two certificates issued by the CA to the RA.
   b. Right-click the CA server in the navigation tree and select **Properties** > **Policy Module**.
   c. Click **Properties**, and then select **Follow the settings in the certificate template, if applicable. Otherwise, automatically issue the certificate**.
4. Modify the Internet information services attributes:
   a. Select **Control Panel** > **Administrative Tools** > **Internet Information Services (IIS) Manager** from the start menu.
   b. Select **Web Sites** from the navigation tree.
   c. Right-click **Default Web Site** and select **Properties** > **Home Directory**.
   d. Specify the path for certificate service in the **Local path** box.
   e. Specify a unique TCP port number for the default website to avoid conflict with existing services. In this example, port 8080 is used.

Configuring the device

1. Synchronize the system time of the device with the CA server for the device to correctly request certificates or obtain CRLs. (Details not shown.)
2. Assign IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

3. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 3.3.3.1 24 2.2.2.2
```

4. Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

5. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones:

   # Configure a rule named **pkilocalout** to allow the device to send packets to the Windows Server 2003 CA server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name pkilocalout
   [Device-security-policy-ip-1-pkilocalout] source-zone local
   [Device-security-policy-ip-1-pkilocalout] destination-zone untrust
   [Device-security-policy-ip-1-pkilocalout] source-ip-host 2.2.2.1
   [Device-security-policy-ip-1-pkilocalout] destination-ip-host 3.3.3.1
   [Device-security-policy-ip-1-pkilocalout] action pass
   [Device-security-policy-ip-1-pkilocalout] quit
   ```

   # Configure a rule named **pkilocalin** to allow the device to receive the packets sent from the Windows Server 2003 CA server.

   ```
   [Device-security-policy-ip] rule name pkilocalin
   [Device-security-policy-ip-2-pkilocalin] source-zone untrust
   [Device-security-policy-ip-2-pkilocalin] destination-zone local
   [Device-security-policy-ip-2-pkilocalin] source-ip-host 3.3.3.1
   [Device-security-policy-ip-2-pkilocalin] destination-ip-host 2.2.2.1
   [Device-security-policy-ip-2-pkilocalin] action pass
   [Device-security-policy-ip-2-pkilocalin] quit
   ```

6. Create an entity named **aaa** and set the common name to **test**.

   ```
   [Device] pki entity aaa
   [Device-pki-entity-aaa] common-name test
   [Device-pki-entity-aaa] country CN
   [Device-pki-entity-aaa] locality pukras
   [Device-pki-entity-aaa] organization abc
   [Device-pki-entity-aaa] quit
   ```

7. Configure a PKI domain and configure parameters for PKI certificate request.

   # Create a PKI domain named **winserver** and enter its view.

   ```
   [Device] pki domain winserver
   ```

   # Set the name of the trusted CA to **myca**.

   ```
   [Device-pki-domain-winserver] ca identifier myca
   ```

   # Configure the certificate request URL. The URL format is **http://***host:port***/certsrv/mscep/mscep.dll**, where *host:port* is the IP address and port number of the CA server.

   ```
   [Device-pki-domain-winserver] certificate request url
   http://4.4.4.1:8080/certsrv/mscep/mscep.dll
   ```

   # Configure the device to send certificate requests to **ra**.

   ```
   [Device-pki-domain-winserver] certificate request from ra
   ```

   # Set the PKI entity name to **aaa**.

   ```
   [Device-pki-domain-winserver] certificate request entity aaa
   ```

   # Configure a general-purpose RSA key pair named **abc** with a length of 1024 bits.

   ```
   [Device-pki-domain-winserver] public-key rsa general name abc length 1024
   [Device-pki-domain-winserver] quit
   ```

8. Generate the RSA local key pair.

```
[Device] public-key local create rsa name abc
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512,it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.........................++++++
.....................................++++++
Create the key pair successfully.
```

9. Request a local certificate:

# Obtain the CA certificate and save it locally.

```
[Device] pki retrieve-certificate domain winserver ca
The trusted CA's finger print is:
    MD5  fingerprint:766C D2C8 9E46 845B 4DCE 439C 1C1F 83AB
    SHA1 fingerprint:97E5 DDED AB39 3141 75FB DB5C E7F8 D7D7 7C9B 97B4
Is the finger print correct?(Y/N):y
Retrieved the certificates successfully.
```

# Submit a certificate request manually.

```
[Device] pki request-certificate domain winserver
Start to request certificate ...
...
Request certificate of domain winserver successfully
```

Verifying the configuration

# Display information about the local certificate in PKI domain **winserver**.

```
[Device] display pki certificate domain winserver local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
             (Negative)01:03:99:ff:ff:ff:ff:fd:11
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: CN=sec
        Validity
            Not Before: Dec 24 07:09:42 2012 GMT
            Not After : Dec 24 07:19:42 2013 GMT
        Subject: C=CN, L=pukras, O=abc, CN=test
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c3:b5:23:a0:2d:46:0b:68:2f:71:d2:14:e1:5a:
                    55:6e:c5:5e:26:86:c1:5a:d6:24:68:02:bf:29:ac:
                    dc:31:41:3f:5d:5b:36:9e:53:dc:3a:bc:0d:11:fb:
                    d6:7d:4f:94:3c:c1:90:4a:50:ce:db:54:e0:b3:27:
                    a9:6a:8e:97:fb:20:c7:44:70:8f:f0:b9:ca:5b:94:
                    f0:56:a5:2b:87:ac:80:c5:cc:04:07:65:02:39:fc:
                    db:61:f7:07:c6:65:4c:e4:5c:57:30:35:b4:2e:ed:
                    9c:ca:0b:c1:5e:8d:2e:91:89:2f:11:e3:1e:12:8a:
```

```
                    f8:dd:f8:a7:2a:94:58:d9:c7:f8:1a:78:bd:f5:42:
                    51:3b:31:5d:ac:3e:c3:af:fa:33:2c:fc:c2:ed:b9:
                    ee:60:83:b3:d3:e5:8e:e5:02:cf:b0:c8:f0:3a:a4:
                    b7:ac:a0:2c:4d:47:5f:39:4b:2c:87:f2:ee:ea:d0:
                    c3:d0:8e:2c:80:83:6f:39:86:92:98:1f:d2:56:3b:
                    d7:94:d2:22:f4:df:e3:f8:d1:b8:92:27:9c:50:57:
                    f3:a1:18:8b:1c:41:ba:db:69:07:52:c1:9a:3d:b1:
                    2d:78:ab:e3:97:47:e2:70:14:30:88:af:f8:8e:cb:
                    68:f9:6f:07:6e:34:b6:38:6a:a2:a8:29:47:91:0e:
                    25:39
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Key Usage:
                Digital Signature, Non Repudiation, Key Encipherment, Data Encip
herment
            X509v3 Subject Key Identifier:
                C9:BB:D5:8B:02:1D:20:5B:40:94:15:EC:9C:16:E8:9D:6D:FD:9F:34
            X509v3 Authority Key Identifier:
                keyid:32:F1:40:BA:9E:F1:09:81:BD:A8:49:66:FF:F8:AB:99:4A:30:21:9
B

            X509v3 CRL Distribution Points:

                Full Name:
                  URI:file://\\g07904c\CertEnroll\sec.crl

            Authority Information Access:
                CA Issuers - URI:http://gc/CertEnroll/gc_sec.crt
                CA Issuers - URI:file://\\gc\CertEnroll\gc_sec.crt

            1.3.6.1.4.1.311.20.2:
                .0.I.P.S.E.C.I.n.t.e.r.m.e.d.i.a.t.e.O.f.f.l.i.n.e
    Signature Algorithm: sha1WithRSAEncryption
        76:f0:6c:2c:4d:bc:22:59:a7:39:88:0b:5c:50:2e:7a:5c:9d:
        6c:28:3c:c0:32:07:5a:9c:4c:b6:31:32:62:a9:45:51:d5:f5:
        36:8f:47:3d:47:ae:74:6c:54:92:f2:54:9f:1a:80:8a:3f:b2:
        14:47:fa:dc:1e:4d:03:d5:d3:f5:9d:ad:9b:8d:03:7f:be:1e:
        29:28:87:f7:ad:88:1c:8f:98:41:9a:db:59:ba:0a:eb:33:ec:
        cf:aa:9b:fc:0f:69:3a:70:f2:fa:73:ab:c1:3e:4d:12:fb:99:
        31:51:ab:c2:84:c0:2f:e5:f6:a7:c3:20:3c:9a:b0:ce:5a:bc:
        0f:d9:34:56:bc:1e:6f:ee:11:3f:7c:b2:52:f9:45:77:52:fb:
        46:8a:ca:b7:9d:02:0d:4e:c3:19:8f:81:46:4e:03:1f:58:03:
        bf:53:c6:c4:85:95:fb:32:70:e6:1b:f3:e4:10:ed:7f:93:27:
        90:6b:30:e7:81:36:bb:e2:ec:f2:dd:2b:bb:b9:03:1c:54:0a:
        00:3f:14:88:de:b8:92:63:1e:f5:b3:c2:cf:0a:d5:f4:80:47:
        6f:fa:7e:2d:e3:a7:38:46:f6:9e:c7:57:9d:7f:82:c7:46:06:
        7d:7c:39:c4:94:41:bd:9e:5c:97:86:c8:48:de:35:1e:80:14:
        02:09:ad:08
```

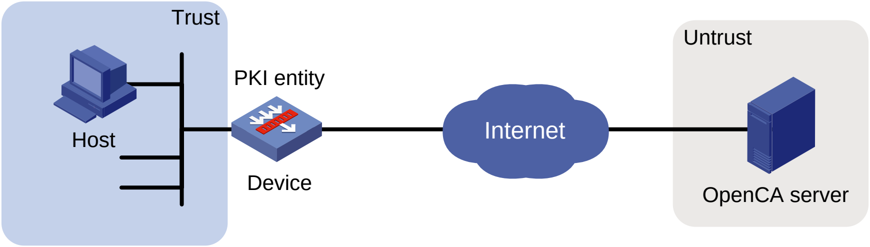
To display detailed information about the CA certificate, use the **display pki certificate domain** command.

Example: Requesting a certificate from an OpenCA server

Network configuration

Configure the PKI entity (the device) to request a local certificate from the OpenCA server.

Figure 5 Network diagram



Configuring the OpenCA server

Configure the OpenCA server as instructed in related manuals. (Details not shown.)

Make sure the version of the OpenCA server is later than version 0.9.2 because earlier versions do not support SCEP.

Configuring the device

1. Synchronize the system time of the device with the CA server for the device to correctly request certificates or obtain CRLs. (Details not shown.)
2. Assign IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

3. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 3.3.3.1 24 2.2.2.2
```

4. Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

5. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones:

   # Configure a rule named **pkilocalout** to allow the device to send packets to the OpenCA server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name pkilocalout
[Device-security-policy-ip-1-pkilocalout] source-zone local
[Device-security-policy-ip-1-pkilocalout] destination-zone untrust
[Device-security-policy-ip-1-pkilocalout] source-ip-host 2.2.2.1
[Device-security-policy-ip-1-pkilocalout] destination-ip-host 3.3.3.1
[Device-security-policy-ip-1-pkilocalout] action pass
[Device-security-policy-ip-1-pkilocalout] quit
```

# Configure a rule named **pkilocalin** to allow the device to receive the packets sent from the OpenCA server.

```
[Device-security-policy-ip] rule name pkilocalin
[Device-security-policy-ip-2-pkilocalin] source-zone untrust
[Device-security-policy-ip-2-pkilocalin] destination-zone local
[Device-security-policy-ip-2-pkilocalin] source-ip-host 3.3.3.1
[Device-security-policy-ip-2-pkilocalin] destination-ip-host 2.2.2.1
[Device-security-policy-ip-2-pkilocalin] action pass
[Device-security-policy-ip-2-pkilocalin] quit
```

6. Create a PKI entity named **aaa** and configure the common name, country code, organization name, and OU for the entity.

```
[Device] pki entity aaa
[Device-pki-entity-aaa] common-name rnd
[Device-pki-entity-aaa] country CN
[Device-pki-entity-aaa] organization test
[Device-pki-entity-aaa] organization-unit software
[Device-pki-entity-aaa] quit
```

7. Configure a PKI domain and configure parameters for PKI certificate request.

# Create a PKI domain named **openca** and enter its view.

```
[Device] pki domain openca
```

# Specify the name of the trusted CA as **myca**.

```
[Device-pki-domain-openca] ca identifier myca
```

# Configure the certificate request URL. The URL is in the format **http://*host*/cgi-bin/pki/scep**, where *host* is the IP address of the OpenCA server.

```
[Device-pki-domain-openca] certificate request url
http://192.168.222.218/cgi-bin/pki/scep
```

# Configure the device to send certificate requests to the RA.

```
[Device-pki-domain-openca] certificate request from ra
```

# Specify PKI entity **aaa** for certificate request.

```
[Device-pki-domain-openca] certificate request entity aaa
```

# Configure a general-purpose RSA key pair named **abc** with a length of 1024 bits.

```
[Device-pki-domain-openca] public-key rsa general name abc length 1024
[Device-pki-domain-openca] quit
```

8. Generate the RSA key pair.

```
[Device] public-key local create rsa name abc
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512,it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..........................++++++
.....................................++++++
Create the key pair successfully.
```

9. Request a local certificate:

# Obtain the CA certificate and save it locally.

```
[Device] pki retrieve-certificate domain openca ca
The trusted CA's finger print is:
    MD5  fingerprint:5AA3 DEFD 7B23 2A25 16A3 14F4 C81C C0FA
    SHA1 fingerprint:9668 4E63 D742 4B09 90E0 4C78 E213 F15F DC8E 9122
Is the finger print correct?(Y/N):y
Retrieved the certificates successfully.
```

# Submit a certificate request manually.

```
[Device] pki request-certificate domain openca
Start to request certificate ...
...
Request certificate of domain openca successfully
```

Verifying the configuration

# Display information about the local certificate in PKI domain **openca**.

```
[Device] display pki certificate domain openca local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            21:1d:b8:d2:e4:a9:21:28:e4:de
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, L=shangdi, ST=pukras, O=OpenCA Labs, OU=mysubUnit, CN=sub-ca,
DC=pki-subdomain, DC=mydomain-sub, DC=com
        Validity
            Not Before: Jun 30 09:09:09 2011 GMT
            Not After : May  1 09:09:09 2012 GMT
        Subject: CN=rnd, O=test, OU=software, C=CN
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:b8:7a:9a:b8:59:eb:fc:70:3e:bf:19:54:0c:7e:
                    c3:90:a5:d3:fd:ee:ff:c6:28:c6:32:fb:04:6e:9c:
                    d6:5a:4f:aa:bb:50:c4:10:5c:eb:97:1d:a7:9e:7d:
                    53:d5:31:ff:99:ab:b6:41:f7:6d:71:61:58:97:84:
                    37:98:c7:7c:79:02:ac:a6:85:f3:21:4d:3c:8e:63:
                    8d:f8:71:7d:28:a1:15:23:99:ed:f9:a1:c3:be:74:
                    0d:f7:64:cf:0a:dd:39:49:d7:3f:25:35:18:f4:1c:
                    59:46:2b:ec:0d:21:1d:00:05:8a:bf:ee:ac:61:03:
                    6c:1f:35:b5:b4:cd:86:9f:45
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Cert Type:
                SSL Client, S/MIME
            X509v3 Key Usage:
                Digital Signature, Non Repudiation, Key Encipherment
            X509v3 Extended Key Usage:
                TLS Web Client Authentication, E-mail Protection, Microsoft
Smartcardlogin
Netscape Comment:
```

```
                User Certificate of OpenCA Labs
            X509v3 Subject Key Identifier:
                24:71:C9:B8:AD:E1:FE:54:9A:EA:E9:14:1B:CD:D9:45:F4:B2:7A:1B
            X509v3 Authority Key Identifier:
                keyid:85:EB:D5:F7:C9:97:2F:4B:7A:6D:DD:1B:4D:DD:00:EE:53:CF:FD:5B

            X509v3 Issuer Alternative Name:
                DNS:root@docm.com, DNS:, IP Address:192.168.154.145, IP
Address:192.168.154.138
            Authority Information Access:
                CA Issuers - URI:http://192.168.222.218/pki/pub/cacert/cacert.crt
                OCSP - URI:http://192.168.222.218:2560/
                1.3.6.1.5.5.7.48.12 - URI:http://192.168.222.218:830/

            X509v3 CRL Distribution Points:

                Full Name:
                  URI:http://192.168.222.218/pki/pub/crl/cacrl.crl

    Signature Algorithm: sha256WithRSAEncryption
        5c:4c:ba:d0:a1:35:79:e6:e5:98:69:91:f6:66:2a:4f:7f:8b:
        0e:80:de:79:45:b9:d9:12:5e:13:28:17:36:42:d5:ae:fc:4e:
        ba:b9:61:f1:0a:76:42:e7:a6:34:43:3e:2d:02:5e:c7:32:f7:
        6b:64:bb:2d:f5:10:6c:68:4d:e7:69:f7:47:25:f5:dc:97:af:
        ae:33:40:44:f3:ab:e4:5a:a0:06:8f:af:22:a9:05:74:43:b6:
        e4:96:a5:d4:52:32:c2:a8:53:37:58:c7:2f:75:cf:3e:8e:ed:
        46:c9:5a:24:b1:f5:51:1d:0f:5a:07:e6:15:7a:02:31:05:8c:
        03:72:52:7c:ff:28:37:1e:7e:14:97:80:0b:4e:b9:51:2d:50:
        98:f2:e4:5a:60:be:25:06:f6:ea:7c:aa:df:7b:8d:59:79:57:
        8f:d4:3e:4f:51:c1:34:e6:c1:1e:71:b5:0d:85:86:a5:ed:63:
        1e:08:7f:d2:50:ac:a0:a3:9e:88:48:10:0b:4a:7d:ed:c1:03:
        9f:87:97:a3:5e:7d:75:1d:ac:7b:6f:bb:43:4d:12:17:9a:76:
        b0:bf:2f:6a:cc:4b:cd:3d:a1:dd:e0:dc:5a:f3:7c:fb:c3:29:
        b0:12:49:5c:12:4c:51:6e:62:43:8b:73:b9:26:2a:f9:3d:a4:
        81:99:31:89
```

To display detailed information about the CA certificate, use the **display pki certificate domain** command.

Example: Configuring IKE negotiation with RSA digital signature from a Windows Server 2003 CA server

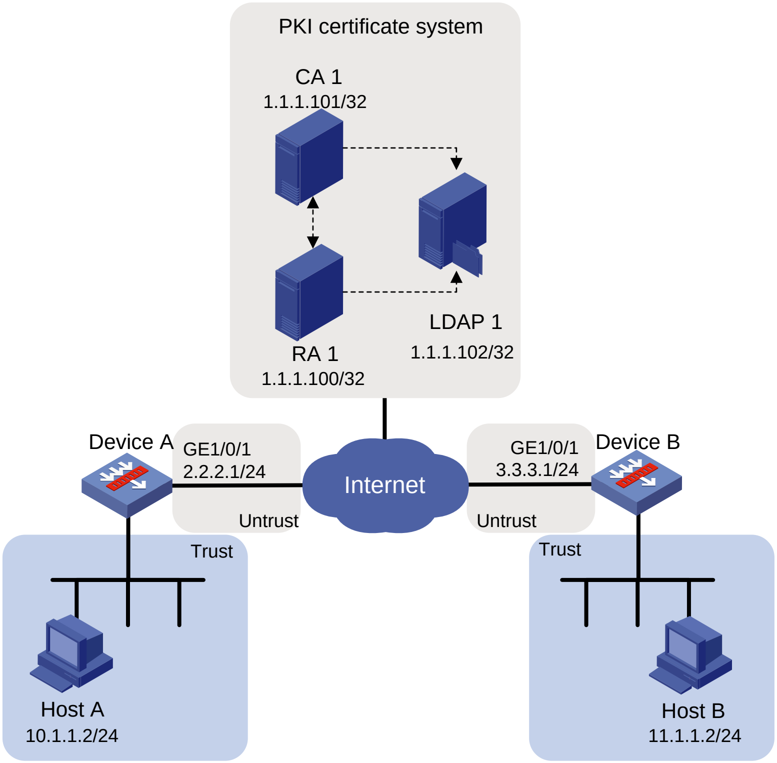
Network configuration

As shown in Figure 6, an IPsec tunnel is established between Device A and Device B to protect the traffic between Host A on subnet 10.1.1.0/24 and Host B on subnet 1.1.1.0/24.

Device A and Device use IKE to set up SAs, and the IKE proposal uses RSA digital signature for identity authentication.

Device A and Device B use the same CA server (Windows Server 2003 CA server).

Figure 6 Network diagram



Configuring the Windows Server 2003 CA server

See "Example: Requesting a certificate from a Windows Server 2003 CA server."

Configuring Device A

# Assign IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 2.2.2.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Configure settings for routing. This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[DeviceA] ip route-static 11.1.1.0 24 2.2.2.2
[DeviceA] ip route-static 3.3.3.1 24 2.2.2.2
```

# Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

# Configure a security policy rule named **pkilocalout** to allow Device A to send IPsec tunnel negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name pkilocalout
[DeviceA-security-policy-ip-1-pkilocalout] source-zone local
[DeviceA-security-policy-ip-1-pkilocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-pkilocalout] source-ip-host 2.2.2.1
[DeviceA-security-policy-ip-1-pkilocalout] destination-ip-host 3.3.3.1
```

```
[DeviceA-security-policy-ip-1-pkilocalout] action pass
[DeviceA-security-policy-ip-1-pkilocalout] quit
```

# Configure a security policy rule named **pkilocalin** to allow Device A to receive the IPsec tunnel negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name pkilocalin
[DeviceA-security-policy-ip-2-pkilocalin] source-zone untrust
[DeviceA-security-policy-ip-2-pkilocalin] destination-zone local
[DeviceA-security-policy-ip-2-pkilocalin] source-ip-host 3.3.3.1
[DeviceA-security-policy-ip-2-pkilocalin] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-pkilocalin] action pass
[DeviceA-security-policy-ip-2-pkilocalin] quit
```

# Configure a security policy rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule 3 name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 11.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a security policy rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule 4 name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 11.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

# Create a PKI entity and configure the IP address, common name, country code, locality, and organization name for the entity.

```
[DeviceA] pki entity en
[DeviceA-pki-entity-en] ip 2.2.2.1
[DeviceA-pki-entity-en] common-name devicea
[DeviceA-pki-entity-en] country CN
[DeviceA-pki-entity-en] locality pukras
[DeviceA-pki-entity-en] organization abc
[DeviceA-pki-entity-en] quit
```

# Create a PKI domain and configure parameters for PKI certificate request.

```
[DeviceA] pki domain 1
[DeviceA-pki-domain-1] ca identifier CA1
[DeviceA-pki-domain-1] certificate request url http://1.1.1.100/certsrv/mscep/mscep.dll
[DeviceA-pki-domain-1] certificate request entity en
[DeviceA-pki-domain-1] ldap-server host 1.1.1.102
```

# Configure the device to send certificate requests to **ra**.

```
[DeviceA-pki-domain-1] certificate request from ra
```

# Configure a general-purpose RSA key pair named **abc** with a length of 1024 bits.

```
[DeviceA-pki-domain-1] public-key rsa general name abc length 1024
```

```
[DeviceA-pki-domain-1] quit
```

# Generate the RSA key pair.

```
[DeviceA] public-key local create rsa name abc
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512,it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.........................++++++
...................................++++++
Create the key pair successfully.
```

# Obtain the CA certificate and save it locally.

```
[DeviceA] pki retrieve-certificate domain 1 ca
```

# Submit a certificate request manually.

```
[DeviceA] pki request-certificate domain 1
```

# Create IKE proposal 1, and configure the authentication method as RSA digital signature.

```
[DeviceA] ike proposal 1
[DeviceA-ike-proposal-1] authentication-method rsa-signature
[DeviceA-ike-proposal-1] quit
```

# Reference the PKI domain used in IKE negotiation for IKE profile **peer**.

```
[DeviceA] ike profile peer
[DeviceA-ike-profile-peer] certificate domain 1
[DeviceA-ike-profile-peer] quit
```

Configuring Device B

# Assign IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface GigabitEthernet1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 3.3.3.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Configure settings for routing. This example configures static routes, and the next hop in the routes is 3.3.3.2.

```
[DeviceB] ip route-static 10.1.1.0 24 3.3.3.2
[DeviceB] ip route-static 2.2.2.1 24 3.3.3.2
```

# Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

# Configure a security policy rule named **pkilocalout** to allow Device B to send IPsec tunnel negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name pkilocalout
[DeviceB-security-policy-ip-1-pkilocalout] source-zone local
[DeviceB-security-policy-ip-1-pkilocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-pkilocalout] source-ip-host 3.3.3.1
[DeviceB-security-policy-ip-1-pkilocalout] destination-ip-host 2.2.2.1
[DeviceB-security-policy-ip-1-pkilocalout] action pass
```

```
[DeviceB-security-policy-ip-1-pkilocalout] quit
```

# Configure a security policy rule named **pkilocalin** to allow Device B to receive the IPsec tunnel negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name pkilocalin
[DeviceB-security-policy-ip-2-pkilocalin] source-zone untrust
[DeviceB-security-policy-ip-2-pkilocalin] destination-zone local
[DeviceB-security-policy-ip-2-pkilocalin] source-ip-host 2.2.2.1
[DeviceB-security-policy-ip-2-pkilocalin] destination-ip-host 3.3.3.1
[DeviceB-security-policy-ip-2-pkilocalin] action pass
[DeviceB-security-policy-ip-2-pkilocalin] quit
```

# Configure a security policy rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule 3 name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 11.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a security policy rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule 4 name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 11.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

# Create a PKI entity and configure the IP address and common name for the entity.

```
[DeviceB] pki entity en
[DeviceB-pki-entity-en] ip 3.3.3.1
[DeviceB-pki-entity-en] common-name deviceb
[DeviceB-pki-entity-en] quit
```

# Create a PKI domain and configure parameters for PKI certificate request.

```
[DeviceB] pki domain 1
[DeviceB-pki-domain-1] ca identifier CA1
[DeviceB-pki-domain-1] certificate request url http://1.1.1.100/certsrv/mscep/mscep.dll
[DeviceB-pki-domain-1] certificate request entity en
[DeviceB-pki-domain-1] ldap-server host 1.1.1.102
```

# Configure the device to send certificate requests to **ra**.

```
[DeviceB-pki-domain-1] certificate request from ra
```

# Configure a general-purpose RSA key pair named **abc** with a length of 1024 bits.

```
[DeviceB-pki-domain-1] public-key rsa general name abc length 1024
[DeviceB-pki-domain-1] quit
```

# Generate the RSA key pair.

```
[DeviceB] public-key local create rsa name abc
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512,it will take a few minutes.
```

```
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..........................++++++
.....................................++++++
Create the key pair successfully.
```

# Obtain the CA certificate and save it locally.

```
[DeviceB] pki retrieve-certificate domain 1 ca
The trusted CA's finger print is:
    MD5  fingerprint:5C41 E657 A0D6 ECB4 6BD6 1823 7473 AABC
    SHA1 fingerprint:1616 E7A5 D89A 2A99 9419 1C12 D696 8228 87BC C266
Is the finger print correct?(Y/N):y
Retrieved the certificates successfully.
```

# Submit a certificate request manually.

```
[DeviceB] pki request-certificate domain 1
Start to request certificate ...
...
Certificate requested successfully.
```

# Create IKE proposal 1, and configure the authentication method as RSA digital signature.

```
[DeviceB] ike proposal 1
[DeviceB-ike-proposal-1] authentication-method rsa-signature
[DeviceB-ike-proposal-1] quit
```

# Reference the PKI domain used in IKE negotiation for IKE profile **peer**.

```
[DeviceB] ike profile peer
[DeviceB-ike-profile-peer] certificate domain 1
[DeviceB-ike-profile-peer] quit
```

The configurations are for IKE negotiation with RSA digital signature. For information about how to configure IPsec SAs to be set up, see IPsec configuration in *VPN Configuration Guide*.

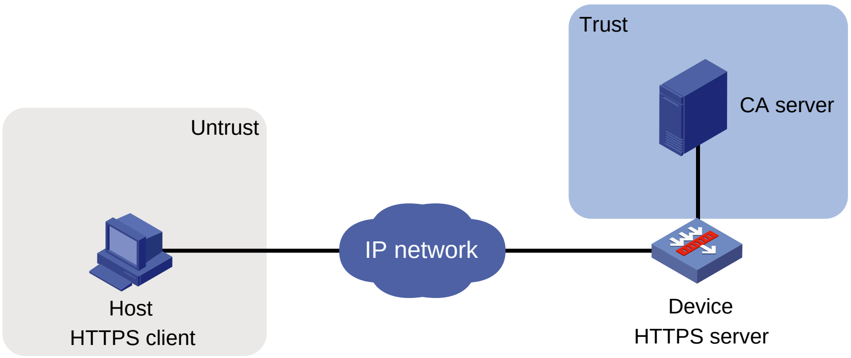
Example: Configuring a certificate-based access control policy

Network configuration

As shown in Figure 7, the host accesses the device through HTTPS.

Configure a certificate-based access control policy on the device to authenticate the host and verify the validity of the host's certificate.

Figure 7 Network diagram



Procedure

1. Create PKI domain **domain1** to be used by SSL. (Details not shown.)
2. Request an SSL server certificate for the device from the CA server. (Details not shown.)
3. Configure the HTTPS server:

# Assign IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 3.3.3.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Add interface GigabitEthernet 1/0/1 to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

# Configure settings for routing. This example configures a static route, and the next hop in the route is 3.3.3.2.

```
[Device] ip route-static 2.2.2.1 24 3.3.3.2
```

# Configure a security policy rule named **pkilocalin** to permit traffic from security zone **Untrust** to security zone **Local**, so that the host can send packets to the device.

```
[Device-security-policy-ip] rule name pkilocalin
[Device-security-policy-ip-1-pkilocalin] source-zone untrust
[Device-security-policy-ip-1-pkilocalin] destination-zone local
[Device-security-policy-ip-1-pkilocalin] source-ip-host 2.2.2.1
[Device-security-policy-ip-1-pkilocalin] destination-ip-host 3.3.3.1
[Device-security-policy-ip-1-pkilocalin] action pass
[Device-security-policy-ip-1-pkilocalin] quit
```

# Configure a security policy rule named **pkilocalout** to permit traffic from security zone **Local** to security zone **Untrust**, so that the device can send packets to the host.

```
[Device-security-policy-ip] rule name pkilocalout
[Device-security-policy-ip-2-pkilocalout] source-zone local
[Device-security-policy-ip-2-pkilocalout] destination-zone untrust
[Device-security-policy-ip-2-pkilocalout] source-ip-host 3.3.3.1
[Device-security-policy-ip-2-pkilocalout] destination-ip-host 2.2.2.1
[Device-security-policy-ip-2-pkilocalout] action pass
[Device-security-policy-ip-2-pkilocalout] quit
```

# Configure an SSL server policy for the HTTPS service.

```
[Device] ssl server-policy abc
```

```
[Device-ssl-server-policy-abc] pki-domain domain1
[Device-ssl-server-policy-abc] client-verify enable
[Device-ssl-server-policy-abc] quit
```

# Apply the SSL server policy to the HTTPS service.

```
[Device] ip https ssl-server-policy abc
```

# Enable the HTTPS service.

```
[Device] ip https enable
```

4. Configure certificate attribute groups:

# Create a certificate attribute group named **mygroup1** and add two attribute rules. The first rule defines that the DN in the subject DN contains the string of **aabbcc**. The second rule defines that the IP address of the certificate issuer is **10.0.0.1**.

```
[Device] pki certificate attribute-group mygroup1
[Device-pki-cert-attribute-group-mygroup1] attribute 1 subject-name dn ctn aabbcc
[Device-pki-cert-attribute-group-mygroup1] attribute 2 issuer-name ip equ 10.0.0.1
[Device-pki-cert-attribute-group-mygroup1] quit
```

# Create a certificate attribute group named **mygroup2** and add two attribute rules. The first rule defines that the FQDN in the alternative subject name does not contain the string of **apple**. The second rule defines that the DN of the certificate issuer name contains the string of **aabbcc**.

```
[Device] pki certificate attribute-group mygroup2
[Device-pki-cert-attribute-group-mygroup2] attribute 1 alt-subject-name fqdn nctn
apple
[Device-pki-cert-attribute-group-mygroup2] attribute 2 issuer-name dn ctn aabbcc
[Device-pki-cert-attribute-group-mygroup2] quit
```

5. Configure a certificate-based access control policy:

# Create a certificate-based access control policy named **myacp**.

```
[Device] pki certificate access-control-policy myacp
```

# Define a statement to deny the certificates that match the attribute rules in the certificate attribute group **mygroup1**.

```
[Device-pki-cert-acp-myacp] rule 1 deny mygroup1
```

# Define a statement to permit the certificates that match the attribute rules in the certificate attribute group **mygroup2**.

```
[Device-pki-cert-acp-myacp] rule 2 permit mygroup2
[Device-pki-cert-acp-myacp] quit
```

# Apply certificate-based access control policy **myacp** to the HTTPS service.

```
[Device] ip https certificate access-control-policy myacp
```

Verifying the configuration

# On the host, access the HTTPS server through a Web browser.

The server first verifies the validity of the host's certificate according to the configured certificate-based access control policy. In the host's certificate, the subject DN is **aabbcc**, the IP address of the certificate issuer is **1.1.1.1**, and the FQDN of the alternative subject name is **banaba**.

The host's certificate does not match the certificate attribute group **mygroup1** specified in **rule 1** of the certificate-based access control policy. The certificate continues to match against rule 2.

The host's certificate matches the certificate attribute group **mygroup2** specified in **rule 2**. Because **rule 2** is a permit statement, the certificate passes the verification and the host can access the HTTPS server.

Example: Importing and exporting certificates

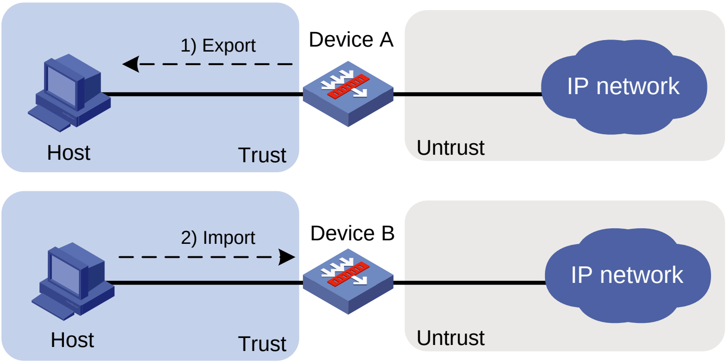
Network configuration

As shown in Figure 8, Device B will replace Device A in the network. The PKI domain **exportdomain** on Device A has two local certificates containing the private key and one CA certificate. To make sure the certificates are still valid after Device B replaces Device A, copy the certificates on Device A to Device B as follows:

1. Export the certificates in PKI domain **exportdomain** on Device A to .pem certificate files.

   During the export, encrypt the private key in the local certificates using 3DES_CBC with the password 11111.
2. Transfer the certificate files from Device A to Device B through FTP.
3. Import the certificate files to PKI domain **importdomain** on Device B.

Figure 8 Network diagram



Procedure

1. Export the certificates on Device A:

   # Assign IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Add interface GigabitEthernet 1/0/1 to security zone **Trust**.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

   # Configure a security policy rule named **pkilocalin** to permit traffic from security zone **Trust** to security zone **Local**, so that the host can send packets to Device A.

```
[DeviceA-security-policy-ip] rule name pkilocalin
[DeviceA-security-policy-ip-1-pkilocalin] source-zone trust
[DeviceA-security-policy-ip-1-pkilocalin] destination-zone local
[DeviceA-security-policy-ip-1-pkilocalin] source-ip-host 192.168.0.2
[DeviceA-security-policy-ip-1-pkilocalin] destination-ip-host 192.168.0.1
[DeviceA-security-policy-ip-1-pkilocalin] action pass
[DeviceA-security-policy-ip-1-pkilocalin] quit
```

   # Configure a security policy rule named **pkilocalout** to permit traffic from security zone **Local** to security zone **Trust**, so that Device A can send packets to the host.

```
[DeviceA-security-policy-ip] rule name pkilocalout
```

```
[DeviceA-security-policy-ip-2-pkilocalout] source-zone local
[DeviceA-security-policy-ip-2-pkilocalout] destination-zone trust
[DeviceA-security-policy-ip-2-pkilocalout] source-ip-host 192.168.0.1
[DeviceA-security-policy-ip-2-pkilocalout] destination-ip-host 192.168.0.2
[DeviceA-security-policy-ip-2-pkilocalout] action pass
[DeviceA-security-policy-ip-2-pkilocalout] quit
```

# Export the CA certificate to a .pem file.

```
[DeviceA] pki export domain exportdomain pem ca filename pkicachain.pem
```

# Export the local certificates to .pem files. Specify the file name as **pkilocal.pem** and use 3DES_CBC to encrypt the private keys with the password **111111**.

```
[DeviceA] pki export domain exportdomain pem local 3des-cbc 111111 filename
pkilocal.pem
```

Now, Device A has three certificate files in PEM format:

- A CA certificate file named **pkicachain.pem**.
- A local certificate file named **pkilocal.pem-signature**, which contains the private key for signature.
- A local certificate file named **pkilocal.pem-encryption**, which contains the private key for encryption.

# Display local certificate file **pkilocal.pem-signature**.

```
[DeviceA] quit
<DeviceA> more pkilocal.pem-signature
Bag Attributes
    friendlyName:
    localKeyID: 90 C6 DC 1D 20 49 4F 24 70 F5 17 17 20 2B 9E AC 20 F3 99 89
subject=/C=CN/O=OpenCA Labs/OU=Users/CN=subsign 11
issuer=/C=CN/L=shangdi/ST=pukras/O=OpenCA Labs/OU=docm/CN=subca1
-----BEGIN CERTIFICATE-----
MIIEgjCCA2qgAwIBAgILAJgsebpejZc5UwAwDQYJKoZIhvcNAQELBQAwZjELMAkG
…
-----END CERTIFICATE-----
Bag Attributes
    friendlyName:
    localKeyID: 90 C6 DC 1D 20 49 4F 24 70 F5 17 17 20 2B 9E AC 20 F3 99 89
Key Attributes: <No Attributes>
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIICxjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQIZtjSjfslJCoCAggA
…
-----END ENCRYPTED PRIVATE KEY-----
```

# Display local certificate file **pkilocal.pem-encryption**.

```
<DeviceA> more pkilocal.pem-encryption
Bag Attributes
    friendlyName:
    localKeyID: D5 DF 29 28 C8 B9 D9 49 6C B5 44 4B C2 BC 66 75 FE D6 6C C8
subject=/C=CN/O=OpenCA Labs/OU=Users/CN=subencr 11
issuer=/C=CN/L=shangdi/ST=pukras/O=OpenCA Labs/OU=docm/CN=subca1
-----BEGIN CERTIFICATE-----
MIIEUDCCAzigAwIBAgIKCHxnAVyzWhIPLzANBgkqhkiG9w0BAQsFADBmMQswCQYD
…
```

```
-----END CERTIFICATE-----
Bag Attributes
    friendlyName:
    localKeyID: D5 DF 29 28 C8 B9 D9 49 6C B5 44 4B C2 BC 66 75 FE D6 6C C8
Key Attributes: <No Attributes>
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIICxjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQI7H0mb4O7/GACAggA
…
-----END ENCRYPTED PRIVATE KEY-----
```

2. Download certificate files **pkicachain.pem**, **pkilocal.pem-signature**, and **pkilocal.pem-encryption** from Device A to the host through FTP. (Details not shown.)
3. Upload the downloaded certificate files (**pkicachain.pem**, **pkilocal.pem-signature**, and **pkilocal.pem-encryption)** from the host to Device B through FTP. (Details not shown.)
4. Import the certificate files to Device B:

# Assign IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.6 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Add interface GigabitEthernet 1/0/1 to security zone **Trust**.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a security policy rule named **pkilocalin** to permit traffic from security zone **Trust** to security zone **Local**, so that the host can send packets to Device B.

```
[DeviceB-security-policy-ip] rule name pkilocalin
[DeviceB-security-policy-ip-1-pkilocalin] source-zone trust
[DeviceB-security-policy-ip-1-pkilocalin] destination-zone local
[DeviceB-security-policy-ip-1-pkilocalin] source-ip-host 192.168.0.5
[DeviceB-security-policy-ip-1-pkilocalin] destination-ip-host 192.168.0.6
[DeviceB-security-policy-ip-1-pkilocalin] action pass
[DeviceB-security-policy-ip-1-pkilocalin] quit
```

# Configure a security policy rule named **pkilocalout** to permit traffic from security zone **Local** to security zone **Trust**, so that Device B can send packets to the host.

```
[DeviceB-security-policy-ip] rule name pkilocalout
[DeviceB-security-policy-ip-2-pkilocalout] source-zone local
[DeviceB-security-policy-ip-2-pkilocalout] destination-zone trust
[DeviceB-security-policy-ip-2-pkilocalout] source-ip-host 192.168.0.6
[DeviceB-security-policy-ip-2-pkilocalout] destination-ip-host 192.168.0.5
[DeviceB-security-policy-ip-2-pkilocalout] action pass
[DeviceB-security-policy-ip-2-pkilocalout] quit
```

# Disable CRL checking. (You can configure CRL checking as required. This example assumes CRL checking is not required.)

```
[DeviceB] pki domain importdomain
[DeviceB-pki-domain-importdomain] undo crl check enable
```

# Specify the RSA key pair for signature as **sign**, and the RSA key pair for encryption as **encr** for certificate request.

```
[DeviceB-pki-domain-importdomain] public-key rsa signature name sign encryption name
encr
```

```
[DeviceB-pki-domain-importdomain] quit
```

# Import CA certificate file **pkicachain.pem** in PEM format to the PKI domain.

```
[DeviceB] pki import domain importdomain pem ca filename pkicachain.pem
```

# Import local certificate file **pkilocal.pem-signature** in PEM format to the PKI domain. The certificate file contains a key pair.

```
[DeviceB] pki import domain importdomain pem local filename pkilocal.pem-signature
Please input the password:******
```

# Import local certificate file **pkilocal.pem-encryption** in PEM format to the PKI domain. The certificate file contains a key pair.

```
[DeviceB] pki import domain importdomain pem local filename pkilocal.pem-encryption
Please input the password:******
```

Verifying the configuration

# Display the imported local certificate information on Device B.

```
[DeviceB] display pki certificate domain importdomain local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            98:2c:79:ba:5e:8d:97:39:53:00
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, L=shangdi, ST=pukras, O=OpenCA Labs, OU=docm, CN=subca1
        Validity
            Not Before: May 26 05:56:49 2011 GMT
            Not After : Nov 22 05:56:49 2012 GMT
        Subject: C=CN, O=OpenCA Labs, OU=Users, CN=subsign 11
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:9f:6e:2f:f6:cb:3d:08:19:9a:4a:ac:b4:ac:63:
                    ce:8d:6a:4c:3a:30:19:3c:14:ff:a9:50:04:f5:00:
                    ee:a3:aa:03:cb:b3:49:c4:f8:ae:55:ee:43:93:69:
                    6c:bf:0d:8c:f4:4e:ca:69:e5:3f:37:5c:83:ea:83:
                    ad:16:b8:99:37:cb:86:10:6b:a0:4d:03:95:06:42:
                    ef:ef:0d:4e:53:08:0a:c9:29:dd:94:28:02:6e:e2:
                    9b:87:c1:38:2d:a4:90:a2:13:5f:a4:e3:24:d3:2c:
                    bf:98:db:a7:c2:36:e2:86:90:55:c7:8c:c5:ea:12:
                    01:31:69:bf:e3:91:71:ec:21
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Cert Type:
                SSL Client, S/MIME
            X509v3 Key Usage:
                Digital Signature, Non Repudiation
            X509v3 Extended Key Usage:
                TLS Web Client Authentication, E-mail Protection, Microsoft
Smartcardlogin
            Netscape Comment:
                User Certificate of OpenCA Labs
            X509v3 Subject Key Identifier:
                AA:45:54:29:5A:50:2B:89:AB:06:E5:BD:0D:07:8C:D9:79:35:B1:F5
            X509v3 Authority Key Identifier:
                keyid:70:54:40:61:71:31:02:06:8C:62:11:0A:CC:A5:DB:0E:7E:74:DE:DD

            X509v3 Subject Alternative Name:
                email:subsign@docm.com
            X509v3 Issuer Alternative Name:
                DNS:subca1@docm.com, DNS:, IP Address:1.1.2.2, IP Address:2.2.1.1
            Authority Information Access:
                CA Issuers - URI:http://titan/pki/pub/cacert/cacert.crt
                OCSP - URI:http://titan:2560/
                1.3.6.1.5.5.7.48.12 - URI:http://titan:830/

            X509v3 CRL Distribution Points:

                Full Name:
                  URI:http://192.168.40.130/pki/pub/crl/cacrl.crl

    Signature Algorithm: sha256WithRSAEncryption
        18:e7:39:9a:ad:84:64:7b:a3:85:62:49:e5:c9:12:56:a6:d2:
        46:91:53:8e:84:ba:4a:0a:6f:28:b9:43:bc:e7:b0:ca:9e:d4:
        1f:d2:6f:48:c4:b9:ba:c5:69:4d:90:f3:15:c4:4e:4b:1e:ef:
        2b:1b:2d:cb:47:1e:60:a9:0f:81:dc:f2:65:6b:5f:7a:e2:36:
        29:5d:d4:52:32:ef:87:50:7c:9f:30:4a:83:de:98:8b:6a:c9:
        3e:9d:54:ee:61:a4:26:f3:9a:40:8f:a6:6b:2b:06:53:df:b6:
        5f:67:5e:34:c8:c3:b5:9b:30:ee:01:b5:a9:51:f9:b1:29:37:
        02:1a:05:02:e7:cc:1c:fe:73:d3:3e:fa:7e:91:63:da:1d:f1:
        db:28:6b:6c:94:84:ad:fc:63:1b:ba:53:af:b3:5d:eb:08:b3:
        5b:d7:22:3a:86:c3:97:ef:ac:25:eb:4a:60:f8:2b:a3:3b:da:
        5d:6f:a5:cf:cb:5a:0b:c5:2b:45:b7:3e:6e:39:e9:d9:66:6d:
        ef:d3:a0:f6:2a:2d:86:a3:01:c4:94:09:c0:99:ce:22:19:84:
        2b:f0:db:3e:1e:18:fb:df:56:cb:6f:a2:56:35:0d:39:94:34:
        6d:19:1d:46:d7:bf:1a:86:22:78:87:3e:67:fe:4b:ed:37:3d:
        d6:0a:1c:0b

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            08:7c:67:01:5c:b3:5a:12:0f:2f
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, L=shangdi, ST=pukras, O=OpenCA Labs, OU=docm, CN=subca1
        Validity
            Not Before: May 26 05:58:26 2011 GMT
            Not After : Nov 22 05:58:26 2012 GMT
        Subject: C=CN, O=OpenCA Labs, OU=Users, CN=subencr 11
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:db:26:13:d3:d1:a4:af:11:f3:6d:37:cf:d0:d4:
                    48:50:4e:0f:7d:54:76:ed:50:28:c6:71:d4:48:ae:
                    4d:e7:3d:23:78:70:63:18:33:f6:94:98:aa:fa:f6:
                    62:ed:8a:50:c6:fd:2e:f4:20:0c:14:f7:54:88:36:
                    2f:e6:e2:88:3f:c2:88:1d:bf:8d:9f:45:6c:5a:f5:
                    94:71:f3:10:e9:ec:81:00:28:60:a9:02:bb:35:8b:
                    bf:85:75:6f:24:ab:26:de:47:6c:ba:1d:ee:0d:35:
                    75:58:10:e5:e8:55:d1:43:ae:85:f8:ff:75:81:03:
                    8c:2e:00:d1:e9:a4:5b:18:39
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Cert Type:
                SSL Server
            X509v3 Key Usage:
                Key Encipherment, Data Encipherment
            Netscape Comment:
                VPN Server of OpenCA Labs
            X509v3 Subject Key Identifier:
                CC:96:03:2F:FC:74:74:45:61:38:1F:48:C0:E8:AA:18:24:F0:2B:AB
            X509v3 Authority Key Identifier:
                keyid:70:54:40:61:71:31:02:06:8C:62:11:0A:CC:A5:DB:0E:7E:74:DE:DD

            X509v3 Subject Alternative Name:
                email:subencr@docm.com
            X509v3 Issuer Alternative Name:
                DNS:subca1@docm.com, DNS:, IP Address:1.1.2.2, IP Address:2.2.1.1
            Authority Information Access:
                CA Issuers - URI:http://titan/pki/pub/cacert/cacert.crt
                OCSP - URI:http://titan:2560/
                1.3.6.1.5.5.7.48.12 - URI:http://titan:830/

            X509v3 CRL Distribution Points:

                Full Name:
                  URI:http://192.168.40.130/pki/pub/crl/cacrl.crl

    Signature Algorithm: sha256WithRSAEncryption
        53:69:66:5f:93:f0:2f:8c:54:24:8f:a2:f2:f1:29:fa:15:16:
        90:71:e2:98:e3:5c:c6:e3:d4:5f:7a:f6:a9:4f:a2:7f:ca:af:
        c4:c8:c7:2c:c0:51:0a:45:d4:56:e2:81:30:41:be:9f:67:a1:
            23:a6:09:50:99:a1:40:5f:44:6f:be:ff:00:67:9d:64:98:fb:
            72:77:9e:fd:f2:4c:3a:b2:43:d8:50:5c:48:08:e7:77:df:fb:
            25:9f:4a:ea:de:37:1e:fb:bc:42:12:0a:98:11:f2:d9:5b:60:
            bc:59:72:04:48:59:cc:50:39:a5:40:12:ff:9d:d0:69:3a:5e:
            3a:09:5a:79:e0:54:67:a0:32:df:bf:72:a0:74:63:f9:05:6f:
            5e:28:d2:e8:65:49:e6:c7:b5:48:7d:95:47:46:c1:61:5a:29:
            90:65:45:4a:88:96:e4:88:bd:59:25:44:3f:61:c6:b1:08:5b:
            86:d2:4f:61:4c:20:38:1c:f4:a1:0b:ea:65:87:7d:1c:22:be:
            b6:17:17:8a:5a:0f:35:4c:b8:b3:73:03:03:63:b1:fc:c4:f5:
            e9:6e:7c:11:e8:17:5a:fb:39:e7:33:93:5b:2b:54:72:57:72:
            5e:78:d6:97:ef:b8:d8:6d:0c:05:28:ea:81:3a:06:a0:2e:c3:
            79:05:cd:c3
```

To display detailed information about the CA certificate, use the **`display pki certificate domain`** command.

Troubleshooting PKI configuration

This section provides troubleshooting information for common problems with PKI.

Failed to obtain the CA certificate

Symptom

The CA certificate cannot be obtained.

Analysis

- The network connection is down, for example, because the network cable is damaged or the connectors have bad contact.
- No trusted CA is specified.
- The certificate request URL is incorrect or not specified.
- The system time of the device is not synchronized with the CA server.
- The CA server does not accept the source IP address specified in the PKI domain, or no source IP address is specified.
- The fingerprint of the root CA certificate is illegal.

Solution

1. Fix the network connection problems, if any.
2. Configure the trusted CA and all other required parameters in the PKI domain.
3. Use the **`ping`** command to verify that the CA server is reachable.
4. Synchronize the system time of the device with the CA server.
5. Specify the correct source IP address that the CA server can accept. For the correct settings, contact the CA administrator.
6. Verify the fingerprint of the CA certificate on the CA server.
7. If the problem persists, contact NSFOCUS Support.

Failed to obtain local certificates

Symptom

The local certificates can be obtained.

Analysis

- The network connection is down.
- The PKI domain does not have a CA certificate before you submit the local certificate request.
- The LDAP server is not configured or is incorrectly configured.
- No key pair is specified for certificate request in the PKI domain, or the specified key pair does not match the one contained in the local certificates to the obtained.
- No PKI entity is configured in the PKI domain, or the PKI entity configuration is incorrect.
- CRL checking is enabled, but the PKI domain does not have a CRL and cannot obtain one.
- The CA server does not accept the source IP address specified in the PKI domain, or no source IP address is specified.
- The system time of the device is not synchronized with the CA server.

Solution

1. Fix the network connection problems, if any..
2. Obtain or import the CA certificate.
3. Configure the correct LDAP server parameters.
4. Specify the key pair for certificate request, or remove the existing key pair, specify a new key pair, and submit a local certificate request again.
5. Check the registration policy on the CA or RA, and make sure the attributes of the PKI entity meet the policy requirements.
6. Obtain the CRL from the CRL repository.
7. Specify the correct source IP address that the CA server can accept. For the correct settings, contact the CA administrator.
8. Synchronize the system time of the device with the CA server.
9. If the problem persists, contact NSFOCUS Support.

Failed to request local certificates

Symptom

Local certificate requests cannot be submitted.

Analysis

- The network connection is down, for example, because the network cable is damaged or the connectors have bad contact.
- The PKI domain does not have a CA certificate before the local certificate request is submitted.
- The certificate request URL is incorrect or is not specified.
- The certificate request reception authority is incorrect or is not specified.
- Required PKI entity parameters are not configured or are incorrectly configured.
- No key pair is specified in the PKI domain for certificate request, or the key pair is changed during a certificate request process.
- Exclusive certificate request applications are running in the PKI domain.

- The CA server does not accept the source IP address specified in the PKI domain, or no source IP address is specified.
- The system time of the device is not synchronized with the CA server.

Solution

1. Fix the network connection problems, if any.
2. Obtain or import the CA certificate.
3. Use the **`ping`** command to verify that the registration server is reachable.
4. Use the **`certificate request from`** command to specify the correct certificate request reception authority.
5. Configure the PKI entity parameters as required by the registration policy on the CA or RA.
6. Specify the key pair for certificate request, or remove the existing key pair, specify a new key pair, and submit a local certificate request again.
7. Use the **`pki abort-certificate-request domain`** command to abort the certificate request.
8. Specify the correct source IP address that the CA server can accept. For the correct settings, contact the CA administrator.
9. Synchronize the system time of the device with the CA server.
10. If the problem persists, contact NSFOCUS Support.

Failed to obtain CRLs

Symptom

CRLs cannot be obtained.

Analysis

- The network connection is down, for example, because the network cable is damaged or the connectors have bad contact.
- The PKI domain does not have a CA certificate before you try to obtain CRLs.
- The URL of the CRL repository is not configured and cannot be obtained from the CA certificate or local certificates in the PKI domain.
- The specified URL of the CRL repository is incorrect.
- The device tries to obtain CRLs through SCEP, but it experiences the following problems:
  - The PKI domain does not have local certificates.
  - The key pairs in the certificates have been changed.
  - The PKI domain has incorrect URL for certificate request.
- The CRL repository uses LDAP for CRL distribution. However, the IP address or host name of the LDAP server is neither contained in the CRL repository URL nor configured in the PKI domain.
- The CA does not issue CRLs.
- The CA server does not accept the source IP address specified in the PKI domain, or no source IP address is specified.

Solution

1. Fix the network connection problems, if any.
2. Obtain or import the CA certificate.
3. If the URL of the CRL repository cannot be obtained, verify that the following conditions exist:
   - The URL for certificate request is valid.

- A local certificate has been successfully obtained.
- The local certificate contains a public key that matches the locally stored key pair.

4. Make sure the LDAP server address is contained in the CRL repository URL, or is configured in the PKI domain.
5. Make sure the CA server support publishing CRLs.
6. Specify a correct source IP address that the CA server can accept. For the correct settings, contact the CA administrator.
7. If the problem persists, contact NSFOCUS Support.

Failed to import the CA certificate

Symptom

The CA certificate cannot be imported.

Analysis

- CRL checking is enabled, but the device does not have a CRL in the PKI domain and cannot obtain one.
- The specified format in which the CA certificate file is to be imported does not match actual certificate file format.

Solution

1. Use the **`undo crl check enable`** command to disable CRL checking in the PKI domain.
2. Make sure the format of the imported file is correct.
3. If the problem persists, contact NSFOCUS Support.

Failed to import the local certificate

Symptom

The local certificate cannot be imported.

Analysis

- The PKI domain does not have a CA certificate, and the local certificate file to be imported does not contain the CA certificate chain.
- CRL checking is enabled, but the device does not have a CRL in the PKI domain and cannot obtain one.
- The specified format in which the local certificate file is to be imported does not match actual certificate file format.
- The device and the certificate do not have the local key pair.
- The certificate has been revoked.
- The certificate is out of the validity period.
- The system time is incorrect.

Solution

1. Obtain or import the CA certificate.
2. Use the **`undo crl check enable`** command to disable CRL checking, or obtain the correct CRL before you import certificates.
3. Make sure the format of the file to be imported is correct.
4. Make sure the certificate file contains the private key.

5. Make sure the certificate is not revoked.
6. Make sure the certificate is valid.
7. Configure the correct system time for the device.
8. If the problem persists, contact NSFOCUS Support.

Failed to export certificates

Symptom

Certificates cannot be exported.

Analysis

- The PKI domain does not have local certificates when you export all certificates in PKCS12 format.
- The specified export path does not exist.
- The specified export path is illegal.
- The public key of the local certificate to be exported does not match the public key of the key pair configured in the PKI domain.
- The storage space of the device is full.

Solution

1. Obtain or request local certificates first.
2. Use the **`mkdir`** command to create the required path.
3. Specify a correct export path.
4. Configure the correct key pair in the PKI domain.
5. Clear up the storage space of the device.
6. If the problem persists, contact NSFOCUS Support.

Failed to set the storage path

Symptom

The storage path for certificates or CRLs cannot be set.

Analysis

- The specified storage path does not exist.
- The specified storage path is illegal.
- The storage space of the device is full.

Solution

1. Use the **`mkdir`** command to create the path.
2. Specify a valid storage path for certificates or CRLs.
3. Clear up the storage space of the device.
4. If the problem persists, contact NSFOCUS Support.

Contents

Configuring SSH

About SSH

Secure Shell (SSH) is a network security protocol. Using encryption and authentication, SSH can implement secure remote access and file transfer over an insecure network.

SSH uses the typical client-server model to establish a channel for secure data transfer based on TCP.

SSH includes two versions: SSH1.x and SSH2.0 (hereinafter referred to as SSH1 and SSH2), which are not compatible. SSH2 is better than SSH1 in performance and security.

SSH applications

The device supports the following SSH applications:

- **Secure Telnet**—Stelnet provides secure and reliable network terminal access services. Through Stelnet, a user can securely log in to a remote server. Stelnet can protect devices against attacks, such as IP spoofing and plain text password interception. The device can act as an Stelnet server or an Stelnet client.
- **Secure File Transfer Protocol**—Based on SSH2, SFTP uses SSH connections to provide secure file transfer. The device can act as an SFTP server, allowing a remote user to log in to the SFTP server for secure file management and transfer. The device can also act as an SFTP client, enabling a user to log in from the device to a remote device for secure file transfer.
- **Secure Copy**—Based on SSH2, SCP offers a secure method to copy files. The device can act as an SCP server, allowing a user to log in to the device for file upload and download. The device can also act as an SCP client, enabling a user to log in from the device to a remote device for secure file transfer.
- **NETCONF over SSH**—Based on SSH2, it enables users to securely log in to the device through SSH and perform NETCONF operations on the device through the NETCONF-over-SSH connections. The device can act only as a NETCONF-over-SSH server. For more information about NETCONF, see *Network Management and Monitoring Configuration Guide*.

When acting as an SSH server or client, the device supports the following SSH versions:

- When acting as an Stelnet, SFTP, or SCP server, the device supports both SSH2 and SSH1.
- When acting as an SSH client, the device supports only SSH2.
- When acting as a NETCONF-over-SSH server, the device supports only SSH2.

How SSH works

This section uses SSH2 as an example to describe the stages to establish an SSH session.

Table 1 Stages to establish an SSH session

Stages	Description
Connection establishment	The SSH server listens to connection requests on port 22. After a client initiates a connection request, the server and the client establish a TCP connection.
Version negotiation	The two parties determine a version to use.
Algorithm negotiation	SSH supports multiple algorithms. Based on the local algorithms, the

Stages	Description
	two parties negotiate the following algorithms: • Key exchange algorithm for generating session keys. • Encryption algorithm for encrypting data. • Public key algorithm for the digital signature and authentication. • HMAC algorithm for protecting data integrity.
Key exchange	The two parties use the DH exchange algorithm to dynamically generate the session keys and session ID. • The session keys are used for protecting data transfer. • The session ID is used for identifying the SSH connection. In this stage, the client also authenticates the server.
Authentication	The SSH server authenticates the client in response to the client's authentication request.
Session request	After passing the authentication, the client sends a session request to the server to request the establishment of a session (or request the Stelnet, SFTP, SCP, or NETCONF service).
Interaction	After the server grants the request, the client and the server start to communicate with each other in the session. In this stage, you can paste commands in text format and execute them at the CLI. The text pasted at one time must be no more than 2000 bytes. As a best practice to ensure the correct execution of commands, paste commands that are in the same view. To execute commands of more than 2000 bytes, save the commands in a configuration file, upload the file to the server through SFTP, and use it to restart the server.

SSH authentication methods

This section describes authentication methods that are supported by the device when it acts as an SSH server.

Password authentication

The SSH server authenticates a client through the AAA mechanism. The password authentication process is as follows:

1. The client sends the server an authentication request that includes the encrypted username and password.
2. The server performs the following operations:
   a. Decrypts the request to get the username and password in plain text.
   b. Verifies the username and password locally or through remote AAA authentication.
   c. Informs the client of the authentication result.

If the remote AAA server requires the user to enter a password for secondary authentication, it send the SSH server an authentication response carrying a prompt. The prompt is transparently transmitted to the client to notify the user to enter a specific password. When the user enters the correct password, the AAA sever examines the password validity. If the password is valid, the SSH server returns an authentication success message to the client.

SSH1 clients do not support secondary password authentication initiated by the AAA server.

For more information about AAA, see "Configuring AAA."

Publickey authentication

The server authenticates a client by verifying the digital signature of the client. The publickey authentication process is as follows:

1. The client sends the server a publickey authentication request that includes the username, public key, and public key algorithm name.

   If the digital certificate of the client is required in authentication, the client also encapsulates the digital certificate in the authentication request. The digital certificate carries the public key information of the client.

2. The server verifies the client's public key.
   - If the public key is invalid, the server informs the client of the authentication failure.
   - If the public key is valid, the server requests the digital signature of the client. After receiving the signature, the server uses the public key to verify the signature and informs the client of the authentication result.

When acting as an SSH server, the device supports using the public key algorithms DSA, ECDSA, and RSA to verify digital signatures.

When acting as an SSH client, the device supports using the public key algorithms DSA, ECDSA, and RSA to generate digital signatures.

For more information about public key configuration, see "Managing public keys."

Password-publickey authentication

The server requires SSH2 clients to pass both password authentication and publickey authentication. However, an SSH1 client only needs to pass either authentication.

Any authentication

The server requires clients to pass password authentication or publickey authentication.

SSH support for Suite B

Suite B contains a set of encryption and authentication algorithms that meet high security requirements. Table 2 lists all algorithms in Suite B.

The SSH server and client support using the X.509v3 certificate for identity authentication in compliance with the algorithm, negotiation, and authentication specifications defined in RFC 6239.

Table 2 Suite B algorithms

Security level	Key exchange algorithm	Encryption algorithm and HMAC algorithm	Public key algorithm
128-bit	ecdh-sha2-nistp256	AES128-GCM	x509v3-ecdsa-sha2-nistp256
192-bit	ecdh-sha2-nistp384	AES256-GCM	x509v3-ecdsa-sha2-nistp384
Both	ecdh-sha2-nistp256 ecdh-sha2-nistp384	AES128-GCM AES256-GCM	x509v3-ecdsa-sha2-nistp256 x509v3-ecdsa-sha2-nistp384

Configuring the device as an SSH server

SSH server tasks at a glance

To configure an SSH server, perform the following tasks:

1. Generating local key pairs

2. (Optional.) Specifying the SSH service port
3. Enabling the SSH server
   - Enabling the Stelnet server
   - Enabling the SFTP server
   - Enabling the SCP server
   - Enabling NETCONF over SSH
4. Configuring the user lines for SSH login

   Required only for Stelnet and NETCONF-over-SSH servers.
5. Configuring a client's host public key

   Required for authentication method **publickey**, **password-publickey,** or **any**.
6. Configuring an SSH user
   - Required for authentication method **publickey**, **password-publickey,** or **any**.
   - Optional for the **password** authentication method.
7. (Optional.) Configuring the SSH management parameters

   SSH management settings, such as authentication and connection control settings, help improve security of SSH connections.
8. (Optional.) Specifying a PKI domain for the SSH server

Generating local key pairs

About this task

The DSA, ECDSA, or RSA key pairs on the SSH server are required for generating the session keys and session ID in the key exchange stage. They can also be used by a client to authenticate the server. When a client authenticates the server, it compares the public key received from the server with the server's public key that the client saved locally. If the keys are consistent, the client uses the locally saved server's public key to decrypt the digital signature received from the server. If the decryption succeeds, the server passes the authentication.

To support SSH clients that use different types of key pairs, generate DSA, ECDSA, and RSA key pairs on the SSH server.

- **RSA key pairs**—The SSH server generates a server key pair and a host key pair for RSA. The RSA server key pair is only used in SSH1 to encrypt the session key for secure transmission of the session key. It is not used in SSH2, because no session key transmission is required in SSH2.
- **DSA key pair**—The SSH server generates only one DSA host key pair. SSH1 does not support the DSA algorithm.
- **ECDSA key pair**—The SSH server generates only one ECDSA host key pair.

Restrictions and guidelines

Local DSA, ECDSA, and RSA key pairs for SSH use default names. You cannot assign names to the key pairs. For more information about creating local key pairs, see public key management commands in *Security Command Reference*.

If the device does not have RSA key pairs with default names, it automatically generates one RSA server key pair and one RSA host key pair when SSH starts. Both key pairs use their default names. The SSH application starts when you execute an SSH server command on the device.

The key modulus length must be less than 2048 bits when you generate the DSA key pair on the SSH server.

When you generate an ECDSA key pair, you can generate only a **secp256r1** or **secp384r1** ECDSA key pair.

Procedure

1. Enter system view.

   system-view

2. Generate local key pairs.

   public-key local create { **dsa** | **ecdsa** { **secp256r1** | **secp384r1** } | **rsa** }

Specifying the SSH service port

About this task

The default port of the SSH service is 22. You can specify another port for the SSH service to improve security of SSH connections.

Procedure

1. Enter system view.

   system-view

2. Specify the SSH service port.

   ssh server port *port-number*

   △ **CAUTION:**

   - If you modify the SSH port number when the SSH server is enabled, the SSH service is restarted and all SSH connections are terminated after the modification. SSH users must reconnect to the SSH server to access the server.
   - If you set the SSH port to a well-known port number, the service that uses the well-known port number might fail to start. Well-known port numbers are in the range of 1 to 1024.

   By default, the SSH service port is 22.

Enabling the Stelnet server

About this task

After you enable the Stelnet server on the device, a client can log in to the device through Stelnet.

Procedure

1. Enter system view.

   system-view

2. Enable the Stelnet server.

   ssh server enable

   By default, the Stelnet server is disabled.

Enabling the SFTP server

About this task

After you enable the SFTP server on the device, a client can log in to the device through SFTP.

Restrictions and guidelines

When acting as an SFTP server, the device does not support SFTP connections initiated by SSH1 clients.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the SFTP server.

   ```
   sftp server enable
   ```

   By default, the SFTP server is disabled.

Enabling the SCP server

About this task

After you enable the SCP server on the device, a client can log in to the device through SCP.

Restrictions and guidelines

When acting as an SCP server, the device does not support SCP connections initiated by SSH1 clients.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the SCP server.

   ```
   scp server enable
   ```

   By default, the SCP server is disabled.

Enabling NETCONF over SSH

About this task

After you enable NETCONF over SSH on the device, a client can perform NETCONF operations on the device through a NETCONF-over-SSH connection.

Restrictions and guidelines

When acting as a server in the NETCONF-over-SSH connection, the device does not support connection requests initiated by SSH1 clients.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable NETCONF over SSH.

   ```
   netconf ssh server enable
   ```

   By default, NETCONF over SSH is disabled.

   For more information about NETCONF over SSH commands, see *Network Management and Monitoring Command Reference*.

Configuring the user lines for SSH login

About this task

Depending on the SSH application, an SSH client can be an Stelnet client, SFTP client, SCP client, or NETCONF-over-SSH client.

Only Stelnet and NETCONF-over-SSH clients require the user line configuration. The user line configuration takes effect on the clients at the next login.

Procedure

1. Enter system view.

   `system-view`

2. Enter VTY user line view.

   `line vty` *`number`* [*`ending-number`*]

3. Set the login authentication mode to scheme.

   `authentication-mode scheme`

   By default, the authentication mode is **password**.

   For more information about this command, see *Fundamentals Command Reference*.

Configuring a client's host public key

About this task

In publickey authentication, the server compares the SSH username and the client's host public key received from the client with the locally saved SSH username and the client's host public key. If they are the same, the server checks the digital signature that the client sends. The client generates the digital signature by using the private key that is paired with the client's host public key.

For publickey authentication, password-publickey authentication, or any authentication, you must perform the following tasks:

1. Configure the client's DSA, ECDSA, or RSA host public key on the server.
2. Specify the associated host private key on the client to generate the digital signature.

   If the device acts as an SSH client, specify the public key algorithm on the client. The algorithm determines the associated host private key for generating the digital signature.

Client public key configuration methods

You can configure the client host public key by using the following methods:

- Manually enter the content of a client's host public key on the server.
  - **a.** Display the host public key on the client and record the key.
  - **b.** Type the client's host public key character by character on the server, or use the copy and paste method.

  The manually entered key must be in DER format without being converted. For the displayed key to meet the requirement when the client is an NSFOCUS device, use the **`display public-key local public`** command. The format of the public key displayed in any other way (for example, by using the **`public-key local export`** command) might be incorrect. If the key is not in correct format, the system discards the key.

- Import the client host public key from a public key file.
  - **a.** Save the client public key file to the server. For example, transfer the client public key file to the server in binary mode through FTP or TFTP.
  - **b.** Import the client public key from the locally saved public key file.

    During the import process, the server automatically converts the host public key to a string in PKCS format.

Restrictions and guidelines

As a best practice, configure no more than 20 SSH client's host public keys on an SSH server.

Import the client's host public key as a best practice.

Entering a client's host public key

1. Enter system view.

   `system-view`

2. Enter public key view.

   `public-key peer` *`keyname`*

3. Configure a client's host public key.

   Enter the content of the client's host public key character by character, or use the copy and paste method.

   When you enter the content of a client's host public key, you can use spaces and carriage returns between characters but the system does not save them. For more information, see "Managing public keys."

4. Exit public key view and save the key.

   `peer-public-key` **end**

Importing a client's host public key from the public key file

1. Enter system view.

   `system-view`

2. Import a client's public key from the public key file.

   `public-key peer` *`keyname`* **`import sshkey`** *`filename`*

Configuring an SSH user

About this task

Configure an SSH user and a local user depending on the authentication method.

- If the authentication method is **publickey**, you must create an SSH user and a local user on the SSH server. The two users must have the same username, so that the SSH user can be assigned the correct working directory and user role.
- If the authentication method is **password**, you must perform one of the following tasks:
  - For local authentication, configure a local user on the SSH server.
  - For remote authentication, configure an SSH user on a remote authentication server, for example, a RADIUS server.

  You do not need to create an SSH user by using the **`ssh user`** command. However, if you want to display all SSH users, including the password-only SSH users, for centralized management, you can use this command to create them. If such an SSH user has been created, make sure you have specified the correct service type and authentication method.
- If the authentication method is **password-publickey** or **any**, you must create an SSH user on the SSH server and perform one of the following tasks:
  - For local authentication, configure a local user on the SSH server.
  - For remote authentication, configure an SSH user on a remote authentication server, for example, a RADIUS server.

  In either case, the local user or the SSH user configured on the remote authentication server must have the same username as the SSH user.

Restrictions and guidelines

If you change the authentication parameters for a logged-in SSH user, the change takes effect on the user at the next login.

For an SFTP or SCP user, the working directory depends on the authentication method.

- If the authentication method is **password**, the working directory is authorized by AAA.

- If the authentication method is **publickey** or **password-publickey**, the working folder is specified by the **`authorization-attribute`** command in the associated local user view.

For an SSH user, the user role also depends on the authentication method.

- If the authentication method is **password**, the user role is authorized by AAA.
- If the authentication method is **publickey** or **password-publickey**, the user role is specified by the **`authorization-attribute`** command in the associated local user view.

For all authentication methods except password authentication, you must specify a client's host public key or digital certificate.

- For a client that sends the user's public key information directly to the server, specify the client's host public key on the server. The specified public key must already exist. For more information about public keys, see "Configuring a client's host public key."
- For a client that sends the user's public key information to the server through a digital certificate, specify the PKI domain on the server. This PKI domain verifies the client's digital certificate. For successful verification, the specified PKI domain must have the correct CA certificate. To specify the PKI domain, use the **`ssh user`** or **`ssh server pki-domain`** command. For more information about configuring a PKI domain, see "Configuring PKI."

For information about configuring local users and remote authentication, see "Configuring AAA."

Procedure

1. Enter system view.

   `system-view`

2. Create an SSH user, and specify the service type and authentication method.

   `ssh user` *`username`* **`service-type`** { **`all`** | **`netconf`** | **`scp`** | **`sftp`** | **`stelnet`** } **`authentication-type`** { **`password`** | { **`any`** | **`password-publickey`** | **`publickey`** } [**`assign`** { **`pki-domain`** *`domain-name`* | **`publickey`** *`keyname`* }] }

   An SSH server supports up to 1024 SSH users.

Configuring the SSH management parameters

Enabling the SSH server to support SSH1 clients

1. Enter system view.

   `system-view`

2. Enable the SSH server to support SSH1 clients.

   `ssh server compatible-ssh1x enable`

   By default, the SSH server does not support SSH1 clients.

Setting the minimum interval for updating the RSA server key pair

1. Enter system view.

   `system-view`

2. Set the minimum interval for updating the RSA server key pair.

   `ssh server rekey-interval` *`interval`*

   By default, the device does not update the RSA server key pair.

   This configuration takes effect only on SSH1 clients.

Setting the SSH user authentication timeout timer

1. Enter system view.

   `system-view`

2. Set the SSH user authentication timeout timer.

ssh server authentication-timeout *time-out-value*

The default setting is 60 seconds.

Perform this task to prevent malicious occupation of TCP connections. If a user does not finish the authentication when the timeout timer expires, the connection cannot be established.

Setting the maximum number of SSH authentication attempts

1. Enter system view.

   system-view

2. Set the maximum number of SSH authentication attempts.

   ssh server authentication-retries *retries*

   The default setting is 3.

   Perform this task to prevent malicious hacking of usernames and passwords. If the authentication method is **any**, the total number of publickey authentication attempts and password authentication attempts cannot exceed the upper limit.

Specifying an SSH login control ACL

1. Enter system view.

   system-view

2. Specify an SSH login control ACL.

   IPv4:

   ssh server acl { *advanced-acl-number* | *basic-acl-number* | **mac** *mac-acl-number* }

   IPv6:

   ssh server ipv6 acl { **ipv6** { *advanced-acl-number* | *basic-acl-number* } | **mac** *mac-acl-number* }

   This feature uses an ACL to filter SSH clients that initiate SSH connections to the server. By default, no ACLs are specified and all SSH users can initiate SSH connections to the server.

Enabling logging for SSH login attempts that are denied by the SSH login control ACL

1. Enter system view.

   system-view

2. Enable logging for SSH login attempts that are denied by the SSH login control ACL.

   ssh server acl-deny-log enable

   By default, logging is disabled for login attempts that are denied by the SSH login control ACL.

   This command enables SSH to generate log messages for SSH login attempts that are denied by the SSH login control ACL and send the messages to the information center.

Setting the DSCP value in the packets that the SSH server sends to SSH clients

1. Enter system view.

   system-view

2. Set the DSCP value in the packets that the SSH server sends to the SSH clients.

   IPv4:

   ssh server dscp *dscp-value*

   IPv6:

   ssh server ipv6 dscp *dscp-value*

   By default, the DSCP value of SSH packets is 48.

   The DSCP value of a packet defines the priority of the packet and affects the transmission priority of the packet. A bigger DSCP value represents a higher priority.

Setting the SFTP connection idle timeout timer

1. Enter system view.

   `system-view`

2. Set the SFTP connection idle timeout timer.

   `sftp server idle-timeout` *`time-out-value`*

   By default, the SFTP connection idle timeout is 10 minutes.

   When the SFTP connection idle timeout timer expires, the system automatically tears the connection down and releases the connection resources.

Setting the maximum number of online SSH users

1. Enter system view.

   `system-view`

2. Set the maximum number of online SSH users.

   `aaa session-limit ssh` *`max-sessions`*

   The default setting is 32.

   When the number of online SSH users reaches the upper limit, the system denies new SSH connection requests. Changing the upper limit does not affect online SSH users.

   For more information about this command, see AAA commands in *Security Command Reference*.

Specifying a PKI domain for the SSH server

About this task

The PKI domain specified for the SSH server has the following functions:

- The SSH server uses the PKI domain to send its certificate to the client in the key exchange stage.
- The SSH server uses the PKI domain to authenticate the client's certificate if no PKI domain is specified for the client authentication by using the **`ssh user`** command.

Procedure

1. Enter system view.

   `system-view`

2. Specify a PKI domain for the SSH server.

   `ssh server pki-domain` *`domain-name`*

   By default, no PKI domain is specified for the SSH server.

Configuring the device as an Stelnet client

Stelnet client tasks at a glance

To configure an Stelnet client, perform the following tasks:

1. Generating local key pairs

   Only required for authentication method **publickey**, **password-publickey**, or **any**.

2. (Optional.) Specifying the source IP address for outgoing SSH packets
3. Establishing a connection to an Stelnet server
4. (Optional.) Establishing a connection to an Stelnet server based on Suite B

Generating local key pairs

About this task

You must generate local key pairs on Stelnet clients when the Stelnet server uses the **publickey**, **password-publickey**, or **any** authentication method.

Restrictions and guidelines

Local DSA, ECDSA, and RSA key pairs for SSH use default names. You cannot assign names to the key pairs. For more information about creating local key pairs, see public key management commands in *Security Command Reference*.

The key modulus length must be less than 2048 bits when you generate a DSA key pair.

When you generate an ECDSA key pair, you can generate only a **secp256r1** or **secp384r1** ECDSA key pair.

Procedure

1. Enter system view.

   `system-view`

2. Generate local key pairs.

   `public-key local create` { **`dsa`** | **`ecdsa`** { **`secp256r1`** | **`secp384r1`** } | **`rsa`** }

Specifying the source IP address for outgoing SSH packets

About this task

After you specify the source IP address for outgoing SSH packets on an Stelnet client, the client uses the specified IP address to communicate with the Stelnet server.

Restrictions and guidelines

As a best practice, specify a loopback interface as the source interface or specify the IP address of a loopback or dialer interface as the source address of outgoing SSH packets for the following purposes:

- Ensuring the communication between the Stelnet client and the Stelnet server.
- Improving the manageability of Stelnet clients in authentication service.

Procedure

1. Enter system view.

   `system-view`

2. Specify the source address for outgoing SSH packets.

   IPv4:

   `ssh client source` { **`interface`** *`interface-type interface-number`* | **`ip`** *`ip-address`* }

   By default, an IPv4 Stelnet client uses the primary IPv4 address of the output interface in the matching route as the source address of the outgoing SSH packets.

   IPv6:

   `ssh client ipv6 source` { **`interface`** *`interface-type interface-number`* | **`ipv6`** *`ipv6-address`* }

   By default, an IPv6 Stelnet client automatically selects a source IPv6 address for outgoing SSH packets in compliance with RFC 3484.

Establishing a connection to an Stelnet server

About this task

Perform this task to enable the Stelnet client feature on the device and establish a connection to the Stelnet server. You can specify the public key algorithm and the preferred encryption, HMAC, and key exchange algorithms to be used during the connection.

To access the server, a client must use the server's host public key to authenticate the server. As a best practice, configure the server's host public key on the device in an insecure network. If the server's host public key is not configured on the client, the client will notify you to confirm whether to continue with the access.

- If you choose to continue, the client accesses the server and downloads the server's host public key. The downloaded public key will be used to authenticate the server in subsequent accesses.
- If you choose to not continue, the connection cannot be established.

Restrictions and guidelines for establishing a connection to an Stelnet server

An Stelnet client cannot establish connections to both IPv4 and IPv6 Stelnet servers.

Establishing a connection to an IPv4 Stelnet server

Execute the following command in user view to establish a connection with an IPv4 Stelnet server:

ssh2 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**identity-key** { **dsa** | **ecdsa-sha2-nistp256** | **ecdsa-sha2-nistp384** | **rsa** | { **x509v3-ecdsa-sha2-nistp256** | **x509v3-ecdsa-sha2-nistp384** } **pki-domain** *domain-name* } | **prefer-compress zlib** | **prefer-ctos-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-ctos-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** } | **prefer-kex** { **dh-group-exchange-sha1** | **dh-group1-sha1** | **dh-group14-sha1** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** } | **prefer-stoc-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-stoc-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** }] * [**dscp** *dscp-value* | **escape** *character* | { **public-key** *keyname* | **server-pki-domain** *domain-name* } | **source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] *

Establishing a connection to an IPv6 Stelnet server

Execute the following command in user view to establish a connection to an IPv6 Stelnet server:

ssh2 ipv6 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**-i** *interface-type interface-number*] [**identity-key** { **dsa** | **ecdsa-sha2-nistp256** | **ecdsa-sha2-nistp384** | **rsa** | { **x509v3-ecdsa-sha2-nistp256** | **x509v3-ecdsa-sha2-nistp384** } **pki-domain** *domain-name* } | **prefer-compress zlib** | **prefer-ctos-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-ctos-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** } | **prefer-kex** { **dh-group-exchange-sha1** | **dh-group1-sha1** | **dh-group14-sha1** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** } | **prefer-stoc-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-stoc-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** }] * [**dscp** *dscp-value* | **escape** *character* | { **public-key** *keyname* | **server-pki-domain** *domain-name* } | **source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] *

Establishing a connection to an Stelnet server based on Suite B

Execute the following command in user view to establish a connection to an Stelnet server based on Suite B:

IPv4:

ssh2 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**] [**dscp** *dscp-value* | **escape** *character* | **source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] *

IPv6:

ssh2 ipv6 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**-i** *interface-type interface-number*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**] [**dscp** *dscp-value* | **escape** *character* | **source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] *

Configuring the device as an SFTP client

SFTP client tasks at a glance

To configure an SFTP client, perform the following tasks:

1. Generating local key pairs

   Only required for authentication method **publickey**, **password-publickey**, or **any**.
2. (Optional.) Specifying the source IP address for outgoing SFTP packets
3. Establishing a connection to an SFTP server
4. (Optional.) Establishing a connection to an SFTP server based on Suite B
5. (Optional.) Working with SFTP directories
6. (Optional.) Working with SFTP files
7. (Optional.) Displaying help information
8. (Optional.) Terminating the connection with the SFTP server

Generating local key pairs

About this task

You must generate local key pairs on SFTP clients when the SFTP server uses the **publickey**, **password-publickey**, or **any** authentication method.

Restrictions and guidelines

Local DSA, ECDSA, and RSA key pairs for SSH use default names. You cannot assign names to the key pairs. For more information about creating local key pairs, see public key management commands in *Security Command Reference*.

The key modulus length must be less than 2048 bits when you generate a DSA key pair.

When you generate an ECDSA key pair, you can generate only a **secp256r1** or **secp384r1** ECDSA key pair.

Procedure

1. Enter system view.

   system-view

2. Generate local key pairs.

   public-key local create { **dsa** | **ecdsa** { **secp256r1** | **secp384r1** } | **rsa** }

Specifying the source IP address for outgoing SFTP packets

About this task

After you specify the source IP address for outgoing SFTP packets on an SFTP client, the client uses the specified IP address to communicate with the SFTP server.

Restrictions and guidelines

As a best practice, specify a loopback interface as the source interface or specify the IP address of a loopback or dialer interface as the source address of outgoing SFTP packets for the following purposes:

- Ensuring the communication between the SFTP client and the SFTP server.
- Improving the manageability of SFTP clients in authentication service.

Procedure

1. Enter system view.

   system-view

2. Specify the source address for outgoing SFTP packets.

   IPv4:

   sftp client source { **ip** *ip-address* | **interface** *interface-type interface-number* }

   By default, an SFTP client uses the primary IPv4 address of the output interface in the matching route as the source address of the outgoing SFTP packets.

   IPv6:

   sftp client ipv6 source { **ipv6** *ipv6-address* | **interface** *interface-type interface-number* }

   By default, an IPv6 SFTP client automatically selects a source IPv6 address for the outgoing SFTP packets in compliance with RFC 3484.

Establishing a connection to an SFTP server

About this task

Perform this task to enable the SFTP client feature on the device and establish a connection to the SFTP server. You can specify the public key algorithm and the preferred encryption, HMAC, and key exchange algorithms to be used during the connection.

To access the server, a client must use the server's host public key to authenticate the server. As a best practice, configure the server's host public key on the device in an insecure network. If the server's host public key is not configured on the client, the client will notify you to confirm whether to continue with the access.

- If you choose to continue, the client accesses the server and downloads the server's host public key. The downloaded public key will be used to authenticate the server in subsequent accesses.
- If you choose to not continue, the connection cannot be established.

Restrictions and guidelines for establishing a connection to an SFTP server

An SFTP client cannot establish connections to both IPv4 and IPv6 SFTP servers.

Establishing a connection to an IPv4 SFTP server

Execute the following command in user view to establish a connection to an IPv4 SFTP server:

sftp *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**identity-key** { **dsa** | **ecdsa-sha2-nistp256** | **ecdsa-sha2-nistp384** | **rsa** | { **x509v3-ecdsa-sha2-nistp256** | **x509v3-ecdsa-sha2-nistp384** } **pki-domain** *domain-name* } | **prefer-compress zlib** | **prefer-ctos-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-ctos-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** } | **prefer-kex** { **dh-group-exchange-sha1** | **dh-group1-sha1** | **dh-group14-sha1** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** } | **prefer-stoc-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-stoc-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** }] * [**dscp** *dscp-value* | { **public-key** *keyname* | **server-pki-domain** *domain-name* } | **source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] *

Establishing a connection to an IPv6 SFTP server

Execute the following command in user view to establish a connection to an IPv6 SFTP server:

sftp ipv6 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**-i** *interface-type interface-number*] [**identity-key** { **dsa** | **ecdsa-sha2-nistp256** | **ecdsa-sha2-nistp384** | **rsa** | { **x509v3-ecdsa-sha2-nistp256** | **x509v3-ecdsa-sha2-nistp384** } **pki-domain** *domain-name* } | **prefer-compress zlib** | **prefer-ctos-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-ctos-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** } | **prefer-kex** { **dh-group-exchange-sha1** | **dh-group1-sha1** | **dh-group14-sha1** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** } | **prefer-stoc-cipher** { **3des-cbc** | **aes128-cbc** | **aes128-ctr** | **aes128-gcm** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** | **aes256-gcm** | **des-cbc** } | **prefer-stoc-hmac** { **md5** | **md5-96** | **sha1** | **sha1-96** | **sha2-256** | **sha2-512** }] * [**dscp** *dscp-value* | { **public-key** *keyname* | **server-pki-domain** *domain-name* } | **source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] *

Establishing a connection to an SFTP server based on Suite B

Execute the following command in user view to establish a connection to an SFTP server based on Suite B:

IPv4:

sftp *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**] [**dscp** *dscp-value* | **source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] *

IPv6:

ssh2 ipv6 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**-i** *interface-type interface-number*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**]

[**dscp** *dscp-value* | **escape** *character* | **source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] *

Working with SFTP directories

About this task

After you establish a connection to an SFTP server, you can operate directories of the SFTP server.

Changing the working directory on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Change the working directory on the SFTP server.

   cd [*remote-path*]
3. (Optional.) Return to the upper-level directory.

   cdup

Displaying the current working directory on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Display the current working directory on the SFTP server.

   pwd

Displaying files under a directory

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Display files under a directory.
   - **dir** [**-a** | **-l**] [*remote-path*]
   - **ls** [**-a** | **-l**] [*remote-path*]

   The **dir** command has the same function as the **ls** command.

Changing the name of a directory on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Change the name of a directory on the SFTP server.

   rename *oldname newname*

Creating a new directory on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Create a new directory on the SFTP server.

   mkdir *remote-path*

Deleting directories on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Delete one or more directories from the SFTP server.

   rmdir *remote-path*

Working with SFTP files

About this task

After you establish a connection to an SFTP server, you can operate files on the SFTP server.

Changing the name of a file on the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Change the name of a file on the SFTP server.

   `rename` *`old-name new-name`*

Downloading a file from the SFTP server and save it locally

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Download a file from the SFTP server and save it locally.

   `get` *`remote-file`* [*`local-file`*]

Uploading a local file to the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Upload a local file to the SFTP server.

   `put` *`local-file`* [*`remote-file`*]

Display files under a directory

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Display files under a directory.
   - **`dir`** [**`-a`** | **`-l`**] [*`remote-path`*]
   - **`ls`** [**`-a`** | **`-l`**] [*`remote-path`*]

   The **`dir`** command has the same function as the **`ls`** command.

Deleting a file from the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Delete a file from the SFTP server.
   - **`delete`** *`remote-file`*
   - **`remove`** *`remote-file`*

   The **`delete`** command has the same function as the **`remove`** command.

Displaying help information

About this task

After you establish a connection to the SFTP server, you can display the help information of SFTP client commands, including the command syntax and parameter configuration.

Procedure

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."

2. Display SFTP client command help information.
   - **help**
   - **?**

   The **help** command has the same function as the **?** command.

Terminating the connection with the SFTP server

1. Enter SFTP client view.

   For more information, see "Establishing a connection to an SFTP server."
2. Terminate the connection with the SFTP server and return to user view.
   - **bye**
   - **exit**
   - **quit**

   The three commands have the same function.

Configuring the device as an SCP client

SCP client tasks at a glance

To configure an SCP client, perform the following tasks:

1. Generating local key pairs

   Only required for the **publickey**, **password-publickey**, or **any** authentication method.
2. Establishing a connection to an SCP server
3. (Optional.) Establishing a connection to an SCP server based on Suite B

Generating local key pairs

About this task

You must generate local key pairs on SCP clients when the SCP server uses the **publickey**, **password-publickey**, or **any** authentication method.

Restrictions and guidelines

Local DSA, ECDSA, and RSA key pairs for SSH use default names. You cannot assign names to the key pairs. For more information about creating local key pairs, see public key management commands in *Security Command Reference*.

The key modulus length must be less than 2048 bits when you generate a DSA key pair.

When you generate an ECDSA key pair, you can generate only a **secp256r1** or **secp384r1** ECDSA key pair.

Procedure

1. Enter system view.

   system-view
2. Generate local key pairs.

   public-key local create { **dsa** | **ecdsa** { **secp256r1** | **secp384r1** } | **rsa** }

Establishing a connection to an SCP server

About this task

Perform this task to enable the SCP client feature on the device, establish a connection to the SCP server, and transfer files with the server. You can specify the public key algorithm and the preferred encryption, HMAC, and key exchange algorithms to be used during the connection.

To access the server, a client must use the server's host public key to authenticate the server. As a best practice, configure the server's host public key on the device in an insecure network. If the server's host public key is not configured on the client, the client will notify you to confirm whether to continue with the access.

- If you choose to continue, the client accesses the server and downloads the server's host public key. The downloaded public key will be used to authenticate the server in subsequent accesses.
- If you choose to not continue, the connection cannot be established.

Restrictions and guidelines for establishing a connection to an SCP server

An SCP client cannot establish connections to both IPv4 and IPv6 SCP servers.

Establishing a connection to an IPv4 SCP server

Execute the following command in user view to connect to an IPv4 SCP server, and transfer files with the server:

```
scp server [ port-number ] [ vpn-instance vpn-instance-name ] { get | put }
source-file-name [ destination-file-name ] [ identity-key { dsa |
ecdsa-sha2-nistp256 | ecdsa-sha2-nistp384 | rsa |
{ x509v3-ecdsa-sha2-nistp256 | x509v3-ecdsa-sha2-nistp384 } pki-domain
domain-name } | prefer-compress zlib | prefer-ctos-cipher { 3des-cbc |
aes128-cbc | aes128-ctr | aes128-gcm | aes192-ctr | aes256-cbc | aes256-ctr
| aes256-gcm | des-cbc } | prefer-ctos-hmac { md5 | md5-96 | sha1 | sha1-96 |
sha2-256 | sha2-512 } | prefer-kex { dh-group-exchange-sha1 |
dh-group1-sha1 | dh-group14-sha1 | ecdh-sha2-nistp256 |
ecdh-sha2-nistp384 } | prefer-stoc-cipher { 3des-cbc | aes128-cbc |
aes128-ctr | aes128-gcm | aes192-ctr | aes256-cbc | aes256-ctr | aes256-gcm
| des-cbc } | prefer-stoc-hmac { md5 | md5-96 | sha1 | sha1-96 | sha2-256 |
sha2-512 } ] * [ { public-key keyname | server-pki-domain domain-name } |
source { interface interface-type interface-number | ip ip-address } ] *
```

Establishing a connection to an IPv6 SCP server

Execute the following command in user view to connect to an IPv6 SCP server, and transfer files with the server.

```
scp ipv6 server [ port-number ] [ vpn-instance vpn-instance-name ] [ -i
interface-type interface-number ] { get | put } source-file-name
[ destination-file-name ] [ identity-key { dsa | ecdsa-sha2-nistp256 |
ecdsa-sha2-nistp384 | rsa | { x509v3-ecdsa-sha2-nistp256 |
x509v3-ecdsa-sha2-nistp384 } pki-domain domain-name } | prefer-compress
zlib | prefer-ctos-cipher { 3des-cbc | aes128-cbc | aes128-ctr | aes128-gcm
| aes192-ctr | aes256-cbc | aes256-ctr | aes256-gcm | des-cbc } |
prefer-ctos-hmac { md5 | md5-96 | sha1 | sha1-96 | sha2-256 | sha2-512 } |
prefer-kex { dh-group-exchange-sha1 | dh-group1-sha1 | dh-group14-sha1 |
ecdh-sha2-nistp256 | ecdh-sha2-nistp384 } | prefer-stoc-cipher { 3des-cbc
| aes128-cbc | aes128-ctr | aes128-gcm | aes192-ctr | aes256-cbc |
aes256-ctr | aes256-gcm | des-cbc } | prefer-stoc-hmac { md5 | md5-96 | sha1
| sha1-96 | sha2-256 | sha2-512 } ] * [ { public-key keyname |
server-pki-domain domain-name } | source { interface interface-type
interface-number | ipv6 ipv6-address } ] *
```

Establishing a connection to an SCP server based on Suite B

Execute the following command in user view to establish a connection to an SCP server based on Suite B:

IPv4:

scp *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] { **get** | **put** } *source-file-name* [*destination-file-name*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**] [**source** { **interface** *interface-type interface-number* | **ip** *ip-address* }] *

IPv6:

scp ipv6 *server* [*port-number*] [**vpn-instance** *vpn-instance-name*] [**-i** *interface-type interface-number*] { **get** | **put** } *source-file-name* [*destination-file-name*] **suite-b** [**128-bit** | **192-bit**] **pki-domain** *domain-name* [**server-pki-domain** *domain-name*] [**prefer-compress zlib**] [**source** { **interface** *interface-type interface-number* | **ipv6** *ipv6-address* }] *

Specifying algorithms for SSH2

About algorithms for SSH2

The SSH2 client and server use the following types of algorithms for algorithm negotiation during the Stelnet, SFTP, or SCP session establishment:

- Key exchange algorithms.
- Public key algorithms.
- Encryption algorithms.
- MAC algorithms.

If you specify algorithms, SSH2 uses only the specified algorithms for algorithm negotiation. The client uses the specified algorithms to initiate the negotiation, and the server uses the matching algorithms to negotiate with the client. If multiple algorithms of the same type are specified, the algorithm specified earlier has a higher priority during negotiation.

Specifying key exchange algorithms for SSH2

1. Enter system view.

   system-view

2. Specify key exchange algorithms for SSH2.

   ssh2 algorithm key-exchange { **dh-group-exchange-sha1** | **dh-group1-sha1** | **dh-group14-sha1** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** } *

   By default, SSH2 uses the **ecdh-sha2-nistp256**, **ecdh-sha2-nistp384**, **dh-group-exchange-sha1**, **dh-group14-sha1**, and **dh-group1-sha1** key exchange algorithms in descending order of priority for algorithm negotiation.

Specifying public key algorithms for SSH2

1. Enter system view.

```
system-view
```

2. Specify public key algorithms for SSH2.

:

```
ssh2 algorithm public-key { dsa | ecdsa-sha2-nistp256 |
ecdsa-sha2-nistp384 | rsa | x509v3-ecdsa-sha2-nistp256 |
x509v3-ecdsa-sha2-nistp384 } *
```

By default, SSH2 uses the **x509v3-ecdsa-sha2-nistp256**, **x509v3-ecdsa-sha2-nistp384**, **ecdsa-sha2-nistp256**, **ecdsa-sha2-nistp384**, **rsa**, and **dsa** public key algorithms in descending order of priority for algorithm negotiation.

Specifying encryption algorithms for SSH2

1. Enter system view.

```
system-view
```

2. Specify encryption algorithms for SSH2.

```
ssh2 algorithm cipher { 3des-cbc | aes128-cbc | aes128-ctr | aes128-gcm
| aes192-ctr | aes256-cbc | aes256-ctr | aes256-gcm | des-cbc } *
```

By default, SSH2 uses the **aes128-ctr**, **aes192-ctr**, **aes256-ctr**, **aes128-gcm**, **aes256-gcm**, **aes128-cbc**, **3des-cbc**, **aes256-cbc**, and **des-cbc** encryption algorithms in descending order of priority for algorithm negotiation.

Specifying MAC algorithms for SSH2

1. Enter system view.

```
system-view
```

2. Specify MAC algorithms for SSH2.

```
ssh2 algorithm mac { md5 | md5-96 | sha1 | sha1-96 | sha2-256 | sha2-512 }
*
```

By default, SSH2 uses the **sha2-256**, **sha2-512, sha1**, **md5**, **sha1-96**, and **md5-96** MAC algorithms in descending order of priority for algorithm negotiation.

Display and maintenance commands for SSH

Execute **display** commands in any view.

Task	Command
Display the public keys of the local key pairs.	**display public-key local** { **dsa** | **ecdsa** | **rsa** | **sm2** } **public** [**name** *publickey-name*]
Display information about peer public keys.	**display public-key peer** [**brief** | **name** *publickey-name*]
Display the source IP address configuration of the SFTP client.	**display sftp client source**
Display the source IP address configuration of the Stelnet client.	**display ssh client source**
Display SSH server status or sessions.	**display ssh server** { **session** [**slot** *slot-number*] | **status** }

Task	Command
Display SSH user information on the SSH server.	**display ssh user-information** [*username*]
Display algorithms used by SSH2 in the algorithm negotiation stage.	**display ssh2 algorithm**

For more information about the **display public-key local** and **display public-key peer** commands, see public key management commands in *Security Command Reference*.

Stelnet configuration examples

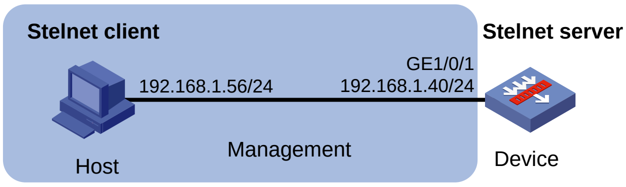
Example: Configuring the device as an Stelnet server (password authentication)

Network configuration

As shown in Figure 1:

- The device acts as the Stelnet server and uses password authentication to authenticate the Stelnet client. The username and password of the client are saved on the device.
- The host acts as the Stelnet client, using Stelnet client software (SSH2). After the user on the host logs in to the device through Stelnet, the user can configure and manage the device as a network administrator.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.40 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name management
   [Device-security-zone-Management] import interface gigabitethernet 1/0/1
   [Device-security-zone-Management] quit
   ```

3. Configure the Stelnet server:

   # Generate RSA key pairs.

   ```
   <Device> system-view
   [Device] public-key local create rsa
   The range of public key modulus is (512 ~ 2048).
   ```

```
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Enable the Stelnet server.

```
[Device] ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
[Device-line-vty0-63] quit
```

# Create a local device management user named **client001**.

```
[Device] local-user client001 class manage
```

# Set the password to **aabbcc** in plain text for local user **client001**.

```
[Device-luser-manage-client001] password simple aabbcc
```

# Authorize local user **client001** to use the **SSH** service.

```
[Device-luser-manage-client001] service-type ssh
```

# Assign the **network-admin** user role to local user **client001**.

```
[Device-luser-manage-client001] authorization-attribute user-role network-admin
[Device-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the service type as **stelnet** and the authentication method as **password** for the user.

```
[Device] ssh user client001 service-type stelnet authentication-type password
```

4. Establish a connection to the Stelnet server:

There are different types of Stelnet client software, such as PuTTY and OpenSSH. This example uses an Stelnet client that runs PuTTY version 0.58.

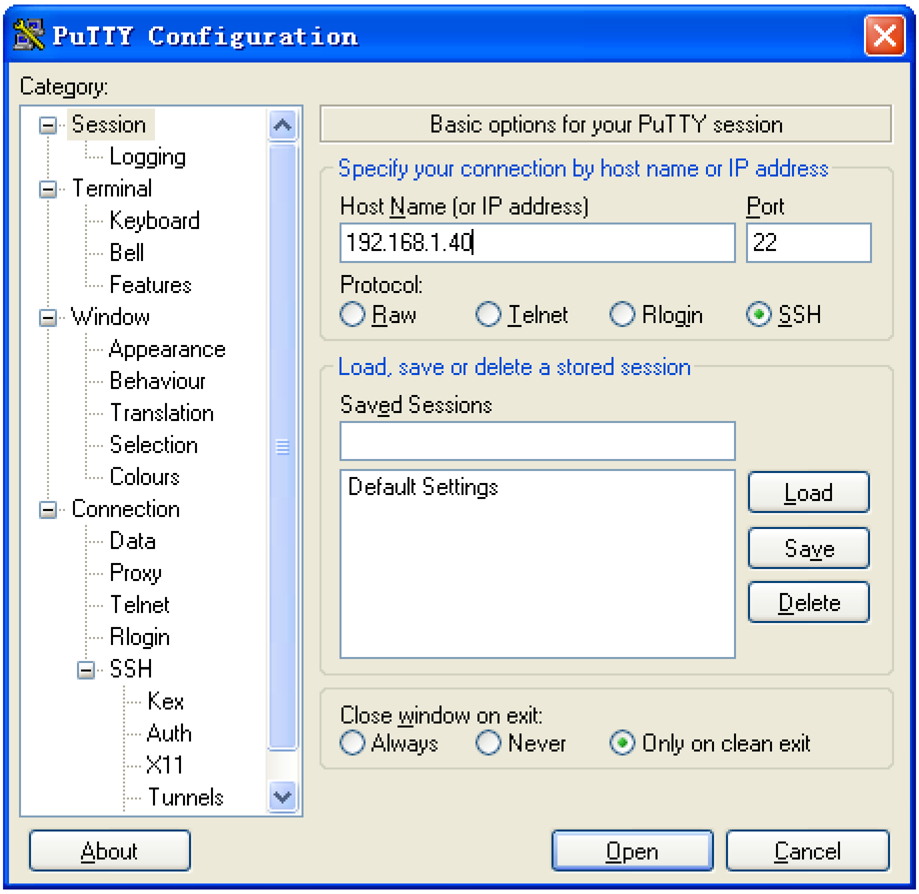
To establish a connection to the Stelnet server:

a. Launch PuTTY.exe to enter the interface shown in Figure 2.

b. In the **Host Name (or IP address)** field, enter the IP address **192.168.1.40** of the Stelnet server.

c. Click **Open**.

Figure 2 Specifying the host name (or IP address)



Host Name (or IP address): 192.168.1.40

Port: 22

Protocol: ○ Raw ○ Telnet ○ Rlogin ● SSH

Close window on exit: ○ Always ○ Never ● Only on clean exit

d. Enter username **client001** and password **aabbcc** to log in to the Stelnet server.

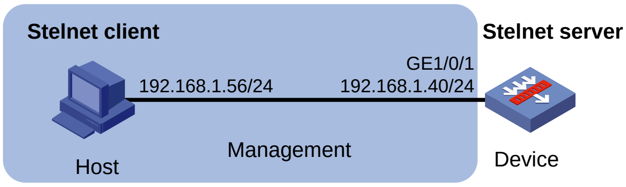
Example: Configuring the device as an Stelnet server (publickey authentication)

Network configuration

As shown in Figure 3:

- The device acts as the Stelnet server, and it uses publickey authentication and the RSA public key algorithm.
- The host acts as the Stelnet client, using Stelnet client software (SSH2). After the user on the host logs in to the device through Stelnet, the user can configure and manage the device as a network administrator.

Figure 3 Network diagram



Procedure

In the server configuration, the client's host public key is required. Use the client software to generate RSA key pairs on the client before configuring the Stelnet server.

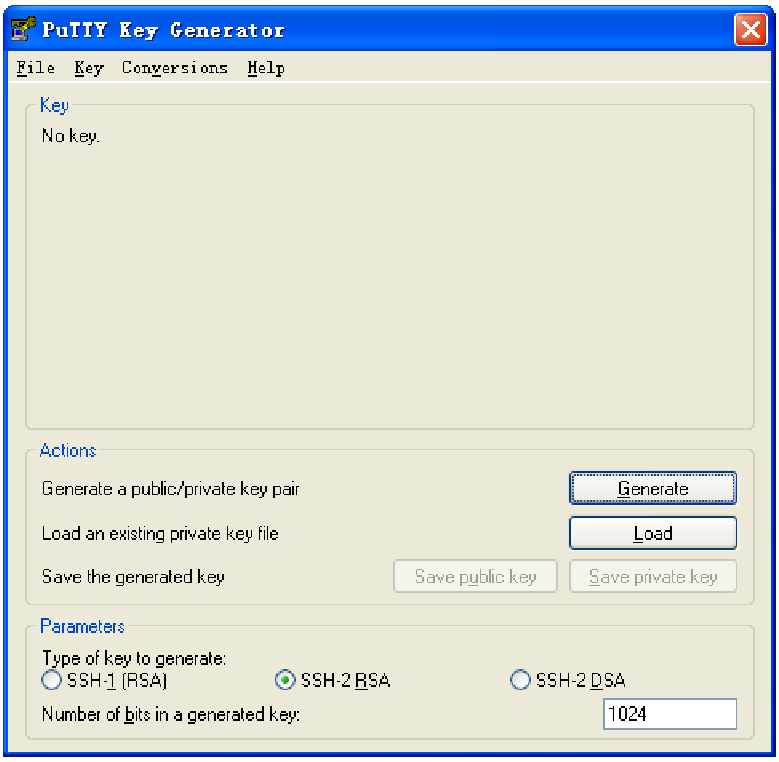
There are different types of Stelnet client software, such as PuTTY and OpenSSH. This example uses an Stelnet client that runs PuTTY version 0.58.

The configuration procedure is as follows:

1. Assign an IP address to GigabitEthernet 1/0/1 on the device. (Details not shown.)
2. Generate RSA key pairs on the Stelnet client:

   a. Run PuTTYGen.exe, select **SSH-2 RSA**, and click **Generate**.

   Figure 4 Generating a key pair on the client



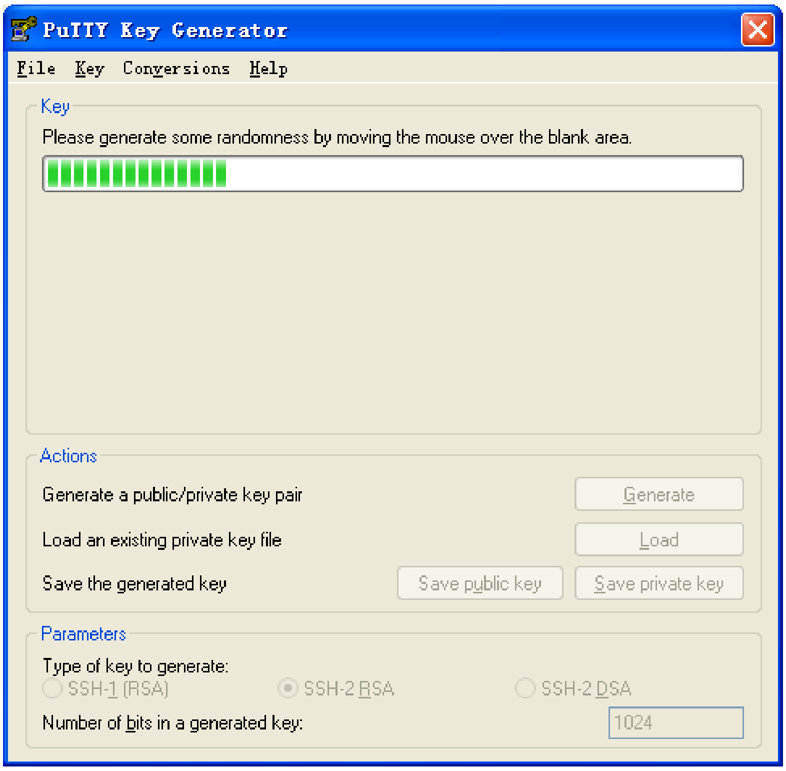
Key
No key.

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

   b. Continue moving the mouse during the key generating process, but do not place the mouse over the green progress bar shown in Figure 5. Otherwise, the progress bar stops moving and the key pair generating process stops.

Figure 5 Generating process



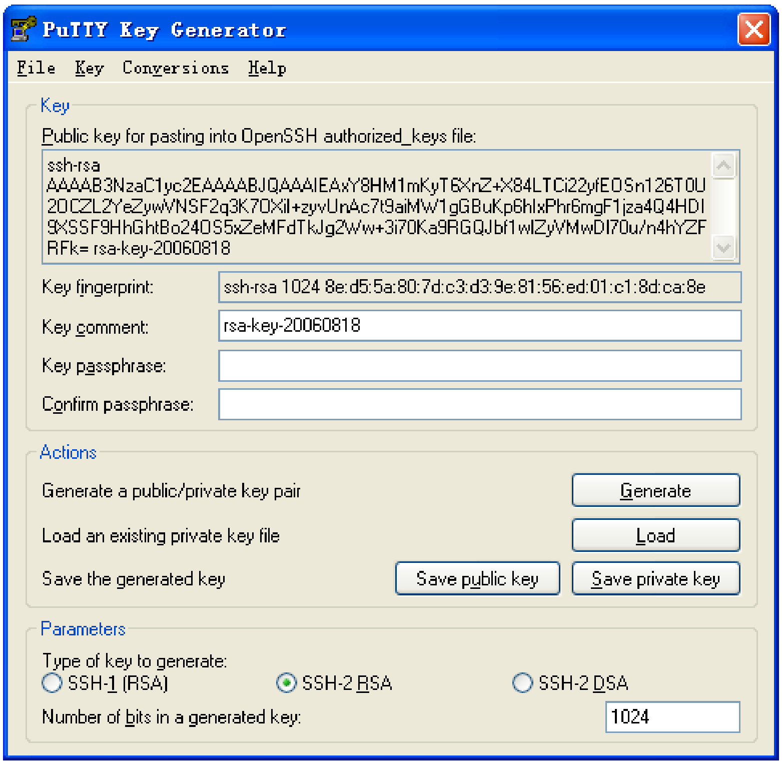
Please generate some randomness by moving the mouse over the blank area.

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

c. After the key pair is generated, click **Save public key** to save the public key. A file saving window appears.

Figure 6 Saving a key pair on the client



Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAIEAxY8HM1mKyT6XnZ+X84LTCi22yfEOSn126T0U
2OCZL2YeZywVNSF2q3K7OXil+zyvUnAc7t9aiMW1gGBuKp6hlxPhr6mgF1jza4Q4HDI
9XSSF9HhGhtBo24OS5xZeMFdTkJg2Ww+3i7OKa9RGQJbf1wlZyVMwDI7Ou/n4hYZF
RFk= rsa-key-20060818
```

Key fingerprint: ssh-rsa 1024 8e:d5:5a:80:7d:c3:d3:9e:81:56:ed:01:c1:8d:ca:8e

Key comment: rsa-key-20060818

Key passphrase:

Confirm passphrase:

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

a. Enter a file name (**key.pub** in this example), and click **Save**.

b. On the page as shown in Figure 6, click **Save private key** to save the private key.

   A confirmation dialog box appears.

c. Click **Yes**.

   A file saving window appears.

d. Enter a file name (**private.ppk** in this example), and click **Save**.

e. Transmit the public key file to the server through FTP or TFTP. (Details not shown.)

3. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.40 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

4. Add interfaces to security zones.

```
[Device] security-zone name management
[Device-security-zone-Management] import interface gigabitethernet 1/0/1
[Device-security-zone-Management] quit
```

5. Configure the Stelnet server:

# Generate RSA key pairs.

```
<Device> system-view
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Enable the Stelnet server.

```
[Device] ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
```

```
[Device-line-vty0-63] quit
```

# Import the peer public key from the public key file **key.pub** and name it **clientkey**.

```
[Device] public-key peer clientkey import sshkey key.pub
```

# Create an SSH user named **client002**. Specify the authentication method as **publickey** for the user, and assign the public key **clientkey** to the user.

```
[Device] ssh user client002 service-type stelnet authentication-type publickey assign publickey clientkey
```

# Create a local device management user named **client002**.

```
[Device] local-user client002 class manage
```

# Authorize local user **client002** to use the **SSH** service.

```
[Device-luser-manage-client002] service-type ssh
```

# Assign the **network-admin** user role to local user **client002**.

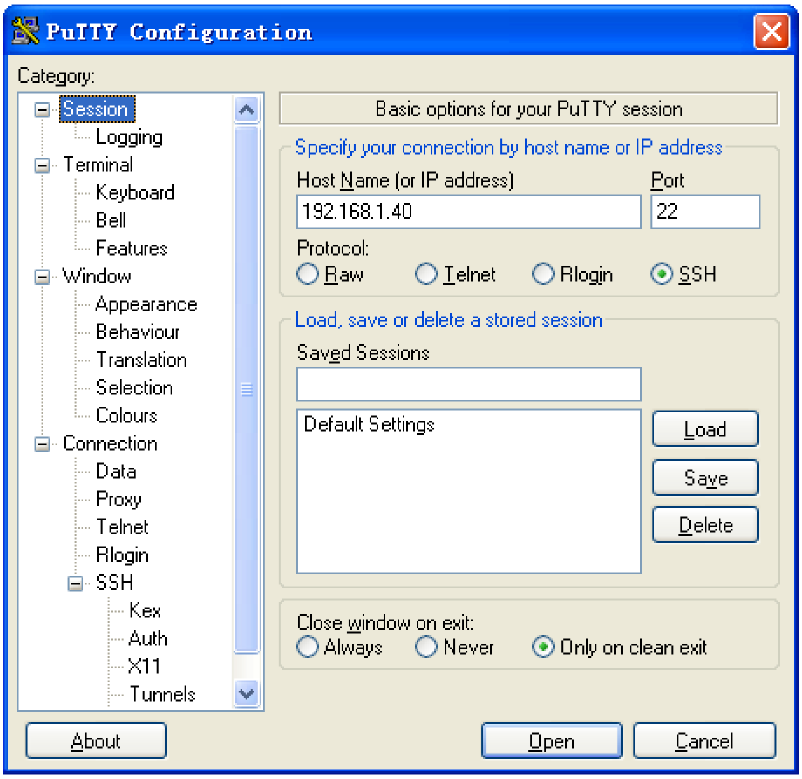
```
[Device-luser-manage-client002] authorization-attribute user-role network-admin
[Device-luser-manage-client002] quit
```

6. Specify the private key file and establish a connection to the Stelnet server:

a. Launch PuTTY.exe on the Stelnet client to enter the interface shown in Figure 7.

b. In the **Host Name (or IP address)** field, enter IP address **192.168.1.40** of the Stelnet server.

Figure 7 Specifying the host name (or IP address)



Host Name (or IP address): 192.168.1.40

Port: 22

Protocol: ○ Raw ○ Telnet ○ Rlogin ● SSH

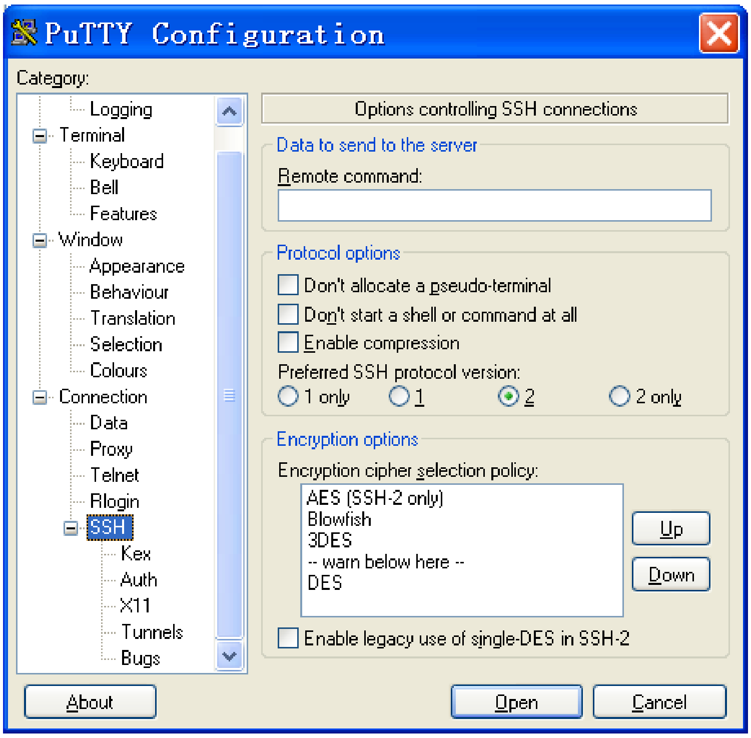
Close window on exit: ○ Always ○ Never ● Only on clean exit

c. From the navigation tree, select **Connection** > **SSH**.

The window shown in Figure 8 appears.

d. Set **Preferred SSH protocol version** to **2**.

Figure 8 Setting the preferred SSH version



Options controlling SSH connections

Data to send to the server

Remote command:

Protocol options

□ Don't allocate a pseudo-terminal

□ Don't start a shell or command at all

□ Enable compression

Preferred SSH protocol version:

○ 1 only ○ 1 ● 2 ○ 2 only

Encryption options

Encryption cipher selection policy:

AES (SSH-2 only)
Blowfish
3DES
-- warn below here --
DES

□ Enable legacy use of single-DES in SSH-2

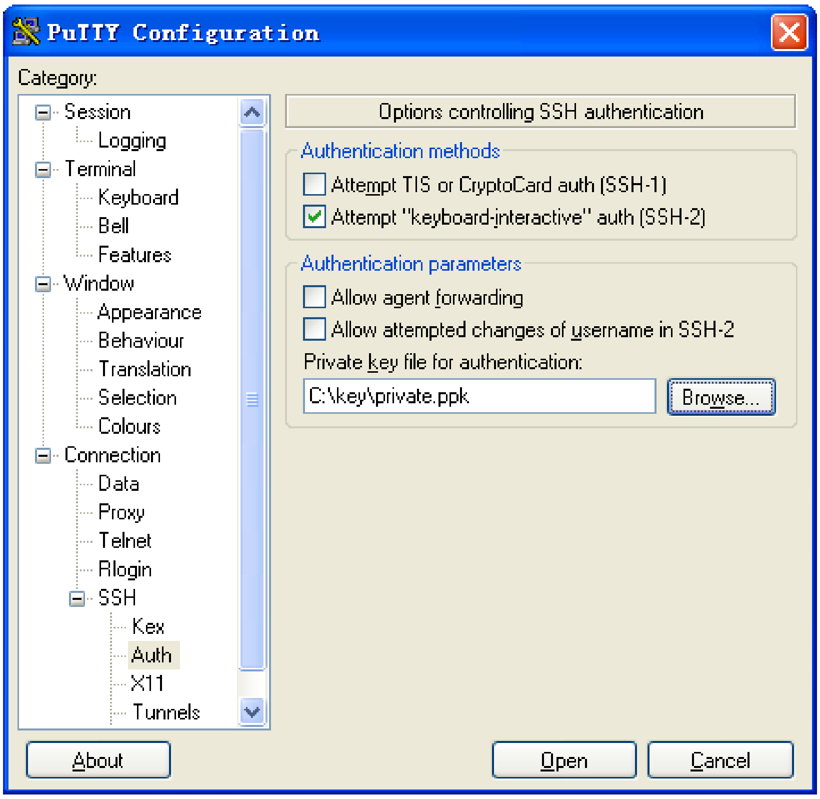
e. From the navigation tree, select **Connection** > **SSH** > **Auth**.

The window shown in Figure 9 appears.

f. Click **Browse…** and select the private key file (**private.ppk** in this example).

g. Click **Open**.

Figure 9 Specifying the private key file



Options controlling SSH authentication

Authentication methods

☐ Attempt TIS or CryptoCard auth (SSH-1)

☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

☐ Allow agent forwarding

☐ Allow attempted changes of username in SSH-2

Private key file for authentication:

C:\key\private.ppk Browse...

a. Enter username **client002** to log in to the Stelnet server.

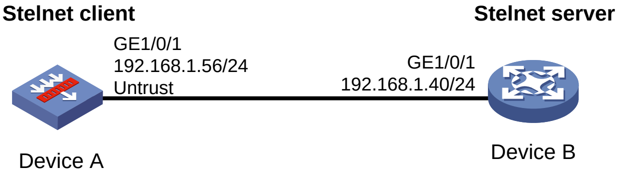
Example: Configuring the device as an Stelnet client (password authentication)

Network configuration

As shown in Figure 10:

- Device B acts as the Stelnet server and uses password authentication to authenticate the Stelnet client. The username and password of the client are saved on Device B.
- Device A acts as the Stelnet client. After the user on Device A logs in to Device B through Stelnet, the user can configure and manage Device B as a network administrator.

Figure 10 Network diagram



Procedure

1. Configure Device A:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
```

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.1.56 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name sshlocalin
[DeviceA-security-policy-ip-1-sshlocalin] source-zone untrust
[DeviceA-security-policy-ip-1-sshlocalin] destination-zone local
[DeviceA-security-policy-ip-1-sshlocalin] source-ip-host 192.168.1.40
[DeviceA-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.1.56
[DeviceA-security-policy-ip-1-sshlocalin] action pass
[DeviceA-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip-1] quit
```

2. Configure Device B:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.1.40 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

3. Configure the Stelnet server:

# Generate RSA key pairs.

```
[DeviceB] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[DeviceB] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[DeviceB] public-key local create ecdsa secp256r1
Generating Keys...
```

```
.
Create the key pair successfully.
```

# Enable the Stelnet server.

```
[DeviceB] ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[DeviceB] line vty 0 63
[DeviceB-line-vty0-63] authentication-mode scheme
[DeviceB-line-vty0-63] quit
```

# Create a local device management user named **client001**.

```
[DeviceB] local-user client001 class manage
```

# Set the password to **aabbcc** in plain text for local user **client001**.

```
[DeviceB-luser-manage-client001] password simple aabbcc
```

# Authorize local user **client001** to use the **SSH** service.

```
[DeviceB-luser-manage-client001] service-type ssh
```

# Assign the **network-admin** user role to local user **client001**.

```
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the service type as **stelnet** and the authentication method as **password** for the user.

```
[DeviceB] ssh user client001 service-type stelnet authentication-type password
```

# Specify public key algorithm **dsa** for SSH2.

```
[DeviceB] ssh2 algorithm public-key dsa
```

4. Establish a connection to the Stelnet server:

- If the client does not have the server's host public key and is connected to the server for the first time:

  # Establish an SSH connection to server 192.168.1.40. Enter username **client001**, and then enter **y** to continue accessing the server without authenticating the server, and enter **y** to download and save the server's host public key.

  ```
  <DeviceA> ssh2 192.168.1.40
  Username: client001
  Press CTRL+C to abort.
  Connecting to 192.168.1.40 port 22.
  The server is not authenticated. Continue? [Y/N]:y
  Do you want to save the server public key? [Y/N]:y
  client001@192.168.1.40's password:
  Enter a character ~ and a dot to abort.

  ******************************************************************************
  * Copyright (c) 2004-2017 NSFOCUS. All rights reserved.                      *
  * Without the owner's prior written consent,                                 *
  * no decompiling or reverse-engineering shall be allowed.                    *
  ******************************************************************************

  <DeviceB>
  ```

  After you enter the correct password, you can access Device B successfully. At the next connection attempt, the client automatically authenticates the server by using the server's host public key that is locally saved on the client, and you only need to enter the password to log in to Device B.

- If you configure the server's host public key on the client before establishing a connection to the server:

  # Use the **display public-key local dsa public** command on the server to display the server's host public key. (Details not shown.)

  # Enter public key view of the client and copy the host public key of the server to the client.

```
<DeviceA> system-view
[DeviceA] public-key peer key1
Enter public key view. Return to system view with "peer-public-key end" command.
[DeviceA-pkey-public-key-key1]
308201B73082012C06072A8648CE3804013082011F0281810
0D757262C4584C44C211F18BD96E5F0
[DeviceA-pkey-public-key-key1]61C4F0A423F7FE6B6B85B34CEF72CE14A0D3A5222FE08CEC
E
65BE6C265854889DC1EDBD13EC8B274
[DeviceA-pkey-public-key-key1]DA9F75BA26CCB987723602787E922BA84421F22C3C89CB9B
0
6FD60FE01941DDD77FE6B12893DA76E
[DeviceA-pkey-public-key-key1]EBC1D128D97F0678D7722B5341C8506F358214B16A2FAC4B
3
68950387811C7DA33021500C773218C
[DeviceA-pkey-public-key-key1]737EC8EE993B4F2DED30F48EDACE915F0281810082269009
E
14EC474BAF2932E69D3B1F18517AD95
[DeviceA-pkey-public-key-key1]94184CCDFCEAE96EC4D5EF93133E84B47093C52B20CD35D0
2
492B3959EC6499625BC4FA5082E22C5
[DeviceA-pkey-public-key-key1]B374E16DD00132CE71B020217091AC717B612391C76C1FB2
E
88317C1BD8171D41ECB83E210C03CC9
[DeviceA-pkey-public-key-key1]B32E810561C21621C73D6DAAC028F4B1585DA7F42519718C
C
9B09EEF0381840002818000AF995917
[DeviceA-pkey-public-key-key1]E1E570A3F6B1C2411948B3B4FFA256699B3BF871221CC9C5
D
F257523777D033BEE77FC378145F2AD
[DeviceA-pkey-public-key-key1]D716D7DB9FCABB4ADBF6FB4FDB0CA25C761B308EF53009F7
1
01F7C62621216D5A572C379A32AC290
[DeviceA-pkey-public-key-key1]E55B394A217DA38B65B77F0185C8DB8095522D1EF044B465
E
8716261214A5A3B493E866991113B2D
[DeviceA-pkey-public-key-key1]485348
[DeviceA-pkey-public-key-key1] peer-public-key end
```

  # Specify public key algorithm **dsa** for SSH2.

```
[DeviceA] ssh2 algorithm public-key dsa
[DeviceA] quit
```

  # Establish an SSH connection to the server, and specify the host public key of the server as **key1**.

```
<DeviceA> ssh2 192.168.1.40 public-key key1
Username: client001
```

```
Press CTRL+C to abort.
Connecting to 192.168.1.40 port 22.
client001@192.168.1.40's password:
Enter a character ~ and a dot to abort.

******************************************************************************
* Copyright (c) 2004-2017 New NSFOCUS. All rights reserved.                  *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<DeviceB>
```

After you enter the correct username and password, you can log in to Device B successfully.

- If the client already has the server's host public key:

```
<DeviceA> ssh2 192.168.1.40
Username: client001
Press CTRL+C to abort.
Connecting to 192.168.1.40 port 22.
client001@192.168.1.40's password:
Enter a character ~ and a dot to abort.

******************************************************************************
* Copyright (c) 2004-2017 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<DeviceB>
```

After you enter the correct username and password, you can access Device B successfully.

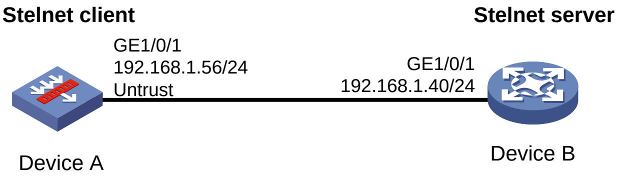
Example: Configuring the device as an Stelnet client (publickey authentication)

Network configuration

As shown in Figure 11:

- Device B acts as the Stelnet server, and it uses publickey authentication and the DSA public key algorithm.
- Device A acts as the Stelnet client. After the user on Device A logs in to Device B through Stelnet, the user can configure and manage Device B as a network administrator.

Figure 11 Network diagram



Procedure

In the server configuration, the client's host public key is required. Generate a DSA key pair on the client before configuring the Stelnet server.

1. Configure Device A:

   a. Assign IP addresses to interfaces:

      # Assign an IP address to interface GigabitEthernet 1/0/1.

      ```
      <DeviceA> system-view
      [DeviceA] interface gigabitethernet 1/0/1
      [DeviceA-GigabitEthernet1/0/1] ip address 192.168.1.56 255.255.255.0
      [DeviceA-GigabitEthernet1/0/1] quit
      ```

      # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

      ```
      [DeviceA] security-zone name Untrust
      [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
      [DeviceA-security-zone-Untrust] quit
      ```

   c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name sshlocalin
      [DeviceA-security-policy-ip-1-sshlocalin] source-zone untrust
      [DeviceA-security-policy-ip-1-sshlocalin] destination-zone local
      [DeviceA-security-policy-ip-1-sshlocalin] source-ip-host 192.168.1.40
      [DeviceA-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.1.56
      [DeviceA-security-policy-ip-1-sshlocalin] action pass
      [DeviceA-security-policy-ip-1-sshlocalin] quit
      [DeviceA-security-policy-ip] quit
      ```

2. Configure Device B:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 192.168.1.40 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

3. Configure the Stelnet client:

   # Generate a DSA key pair.

   ```
   <DeviceA> system-view
   [DeviceA] public-key local create dsa
   The range of public key modulus is (512 ~ 2048).
   If the key modulus is greater than 512, it will take a few minutes.
   Press CTRL+C to abort.
   Input the modulus length [default = 1024]:
   Generating Keys...
   ..................
   Create the key pair successfully.
   ```

   # Export the DSA host public key to a public key file named **key.pub**.

   ```
   [DeviceA] public-key local export dsa ssh2 key.pub
   [DeviceA] quit
   ```

# Transmit the public key file **key.pub** to the server through FTP or TFTP. (Details not shown.)

4. Configure the Stelnet server:

# Generate RSA key pairs.

```
<DeviceB> system-view
[DeviceB] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[DeviceB] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[DeviceB] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Enable the Stelnet server.

```
[DeviceB] ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[DeviceB] line vty 0 63
[DeviceB-line-vty0-63] authentication-mode scheme
[DeviceB-line-vty0-63] quit
```

# Import the peer public key from the public key file **key.pub**, and name it **clientkey**.

```
[DeviceB] public-key peer clientkey import sshkey key.pub
```

# Create an SSH user named **client002**. Specify the authentication method as **publickey** for the user, and assign the public key **clientkey** to the user.

```
[DeviceB] ssh user client002 service-type stelnet authentication-type publickey
assign publickey clientkey
```

# Create a local device management user named **client002**.

```
[DeviceB] local-user client002 class manage
```

# Authorize local user **client002** to use the **SSH** service.

```
[DeviceB-luser-manage-client002] service-type ssh
```

# Assign the **network-admin** user role to local user **client002**.

```
[DeviceB-luser-manage-client002] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client002] quit
```

5. Establish an SSH connection to the Stelnet server.

```
<DeviceA> ssh2 192.168.1.40 identity-key dsa
Username: client002
```

```
Press CTRL+C to abort.
Connecting to 192.168.1.40 port 22.
The server is not authenticated. Continue? [Y/N]:y
Do you want to save the server public key? [Y/N]:n
Enter a character ~ and a dot to abort.

******************************************************************************
* Copyright (c) 2004-2017 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<DeviceB>
```

After you enter username **client002** and then enter **y** to continue accessing the server, you can log in to the server successfully.

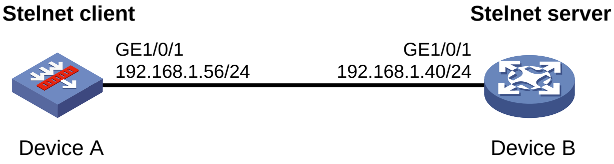
Example: Configuring Stelnet based on 128-bit Suite B algorithms

Network configuration

As shown in Figure 12:

- Device B acts as the Stelnet Suite B server (SSH2), and it uses publickey authentication to authenticate the client.
- Device A acts as an Stelnet Suite B client (SSH2). After the user on Device A logs in to Device B through the Stelnet Suite B client software, the user can configure and manage Device B as an administrator.

Figure 12 Network diagram



Procedure

1. Generate the client's certificate and the server's certificate. (Details not shown.)

   You must first configure the certificates of the server and the client because they are required for identity authentication between the two parties.

   In this example, the server's certificate file is **ssh-server-ecdsa256.p12** and the client's certificate file is **ssh-client-ecdsa256.p12**.

2. Configure Device A:

   a. Assign IP addresses to interfaces:

      # Assign an IP address to interface GigabitEthernet 1/0/1.

      ```
      <DeviceA> system-view
      [DeviceA] interface gigabitethernet 1/0/1
      [DeviceA-GigabitEthernet1/0/1] ip address 192.168.1.56 255.255.255.0
      [DeviceA-GigabitEthernet1/0/1] quit
      ```

      # Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name sshlocalin
[DeviceA-security-policy-ip-1-sshlocalin] source-zone untrust
[DeviceA-security-policy-ip-1-sshlocalin] destination-zone local
[DeviceA-security-policy-ip-1-sshlocalin] source-ip-host 192.168.1.40
[DeviceA-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.1.56
[DeviceA-security-policy-ip-1-sshlocalin] action pass
[DeviceA-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip] quit
```

3. Configure Device B:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.1.40 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

4. Configure the Stelnet client:

You can modify the pkix version of the client software OpenSSH to support Suite B. This example uses an NSFOCUS device as an Stelnet client.

# Upload the server's certificate file **ssh-server-ecdsa256.p12** and the client's certificate file **ssh-client-ecdsa256.p12** to the Stelnet client through FTP or TFTP. (Details not shown.)

# Create a PKI domain named **server256** for verifying the server's certificate and enter its view.

```
<DeviceA> system-view
[DeviceA] pki domain server256
```

# Disable CRL checking.

```
[DeviceA-pki-domain-server256] undo crl check enable
[DeviceA-pki-domain-server256] quit
```

# Import local certificate file **ssh-server-ecdsa256.p12** to PKI domain **server256**.

```
[DeviceA] pki import domain server256 p12 local filename ssh-server-ecdsa256.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: server256]:
```

# Display information about local certificates in PKI domain **server256**.

```
[DeviceA] display pki certificate domain server256 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 3 (0x3)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
```

```
            Not Before: Aug 21 08:39:51 2015 GMT
            Not After : Aug 20 08:39:51 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=SSH Server secp256
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:a2:b4:b4:66:1e:3b:d5:50:50:0e:55:19:8d:52:
                    6d:47:8c:3d:3d:96:75:88:2f:9a:ba:a2:a7:f9:ef:
                    0a:a9:20:b7:b6:6a:90:0e:f8:c6:de:15:a2:23:81:
                    3c:9e:a2:b7:83:87:b9:ad:28:c8:2a:5e:58:11:8e:
                    c7:61:4a:52:51
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                08:C1:F1:AA:97:45:19:6A:DA:4A:F2:87:A1:1A:E8:30:BD:31:30:D7
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA256
         30:65:02:31:00:a9:16:e9:c1:76:f0:32:fc:4b:f9:8f:b6:7f:
         31:a0:9f:de:a7:cc:33:29:27:2c:71:2e:f9:0d:74:cb:25:c9:
         00:d2:52:18:7f:58:3f:cc:7e:8b:d3:42:65:00:cb:63:f8:02:
         30:01:a2:f6:a1:51:04:1c:61:78:f6:6b:7e:f9:f9:42:8d:7c:
         a7:bb:47:7c:2a:85:67:0d:81:12:0b:02:98:bc:06:1f:c1:3c:
         9b:c2:1b:4c:44:38:5a:14:b2:48:63:02:2b
```

# Create a PKI domain named **client256** for the client's certificate and enter its view.

```
[DeviceA] pki domain client256
```

# Disable CRL checking.

```
[DeviceA-pki-domain-client256] undo crl check enable
[DeviceA-pki-domain-client256] quit
```

# Import local certificate file **ssh-client-ecdsa256.p12** to PKI domain **client256**.

```
[DeviceA] pki import domain client256 p12 local filename ssh-client-ecdsa256.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: client256]:
```

# Display information about local certificates in PKI domain **client256**.

```
[DeviceA] display pki certificate domain client256 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 4 (0x4)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 21 08:41:09 2015 GMT
            Not After : Aug 20 08:41:09 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=SSH Client secp256
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:da:e2:26:45:87:7a:63:20:e7:ca:7f:82:19:f5:
                    96:88:3e:25:46:f8:2f:9a:4c:70:61:35:db:e4:39:
                    b8:38:c4:60:4a:65:28:49:14:32:3c:cc:6d:cd:34:
                    29:83:84:74:a7:2d:0e:75:1c:c2:52:58:1e:22:16:
                    12:d0:b4:8a:92
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                1A:61:60:4D:76:40:B8:BA:5D:A1:3C:60:BC:57:98:35:20:79:80:FC
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA256
         30:66:02:31:00:9a:6d:fd:7d:ab:ae:54:9a:81:71:e6:bb:ad:
         5a:2e:dc:1d:b3:8a:bf:ce:ee:71:4e:8f:d9:93:7f:a3:48:a1:
         5c:17:cb:22:fa:8f:b3:e5:76:89:06:9f:96:47:dc:34:87:02:
         31:00:e3:af:2a:8f:d6:8d:1f:3a:2b:ae:2f:97:b3:52:63:b6:
         18:67:70:2c:93:2a:41:c0:e7:fa:93:20:09:4d:f4:bf:d0:11:
         66:0f:48:56:01:1e:c3:be:37:4e:49:19:cf:c6
```

5. Configure the Stelnet server:

   # Upload the server's certificate file **ssh-server-ecdsa256.p12** and the client's certificate file **ssh-client-ecdsa256.p12** to the Stelnet server through FTP or TFTP. (Details not shown.)

   # Create a PKI domain named **client256** for verifying the client's certificate and import the file of the client's certificate to this domain. (Details not shown.)

   # Create a PKI domain named **server256** for the server's certificate and import the file of the server's certificate to this domain. (Details not shown.)

   # Specify Suite B algorithms for algorithm negotiation.

   ```
   <DeviceB> system-view
   [DeviceB] ssh2 algorithm key-exchange ecdh-sha2-nistp256
   [DeviceB] ssh2 algorithm cipher aes128-gcm
   [DeviceB] ssh2 algorithm public-key x509v3-ecdsa-sha2-nistp256
   x509v3-ecdsa-sha2-nistp384
   ```

   # Specify **server256** as the PKI domain of the server's certificate.

```
[DeviceB] ssh server pki-domain server256
```

# Enable the Stelnet server.

```
[DeviceB] ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[DeviceB] line vty 0 63
[DeviceB-line-vty0-63] authentication-mode scheme
[DeviceB-line-vty0-63] quit
```

# Create a local device management user named **client001**. Authorize the user to use the **SSH** service and assign the **network-admin** user role to the user.

```
[DeviceB] local-user client001 class manage
[DeviceB-luser-manage-client001] service-type ssh
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the **publickey** authentication method for the user and specify **client256** as the PKI domain for verifying the client's certificate.

```
[DeviceB] ssh user client001 service-type stelnet authentication-type publickey
assign pki-domain client256
```

6. Establish an SSH connection to the Stelnet server based on the 128-bit Suite B algorithms:

# Establish an SSH connection to the server at 192.168.1.40.

```
<DeviceA> ssh2 192.168.1.40 suite-b 128-bit pki-domain client256 server-pki-domain
server256
Username: client001
Press CTRL+C to abort.
Connecting to 192.168.1.40 port 22.
Enter a character ~ and a dot to abort.

******************************************************************************
* Copyright (c) 2004-2017 NSFOCUS. All rights reserved.                      *
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<DeviceB>
```

SFTP configuration examples

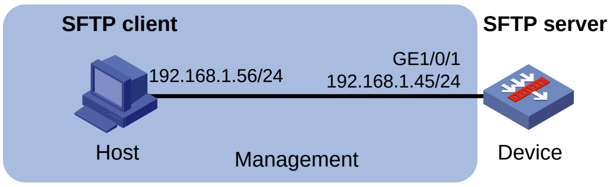
Example: Configuring the device as an SFTP server (password authentication)

Network configuration

As shown in Figure 13:

- The device acts as the SFTP server and uses password authentication to authenticate the SFTP client. The username and password of the client are saved on the device.
- The host acts as the SFTP client. After the user on the client logs in to the device through SFTP, the user can perform file management and transfer operations on the device as a network administrator.

Figure 13 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.45 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name management
   [Device-security-zone-Management] import interface gigabitethernet 1/0/1
   [Device-security-zone-Management] quit
   ```

3. Configure the SFTP server:

   # Generate RSA key pairs.

   ```
   <Device> system-view
   [Device] public-key local create rsa
   The range of public key modulus is (512 ~ 2048).
   If the key modulus is greater than 512, it will take a few minutes.
   Press CTRL+C to abort.
   Input the modulus length [default = 1024]:
   Generating Keys...
   ..................
   Create the key pair successfully.
   ```

   # Generate a DSA key pair.

   ```
   [Device] public-key local create dsa
   The range of public key modulus is (512 ~ 2048).
   If the key modulus is greater than 512, it will take a few minutes.
   Press CTRL+C to abort.
   Input the modulus length [default = 1024]:
   Generating Keys...
   ..................
   Create the key pair successfully.
   ```

   # Generate an ECDSA key pair.

   ```
   [Device] public-key local create ecdsa secp256r1
   Generating Keys...
   .
   Create the key pair successfully.
   ```

   # Enable the SFTP server.

   ```
   [Device] sftp server enable
   ```

   # Create a local device management user named **client002**.

```
[Device] local-user client002 class manage
```

# Set the password to **aabbcc** in plain text for local user **client002**.

```
[Device-luser-manage-client002] password simple aabbcc
```

# Authorize local user **client002** to use the **SSH** service.

```
[Device-luser-manage-client002] service-type ssh
```

# Assign the **network-admin** user role and working directory **flash:/** to local user **client002**.

```
[Device-luser-manage-client002] authorization-attribute user-role network-admin
work-directory flash:/
[Device-luser-manage-client002] quit
```

# Create an SSH user named **client002**. Specify the authentication method as **password** and service type as **sftp** for the user.

```
[Device] ssh user client002 service-type sftp authentication-type password
```

4. Establish a connection to the SFTP server:

   This example uses an SFTP client that runs PSFTP of PuTTY version 0.58. PSFTP supports only password authentication.

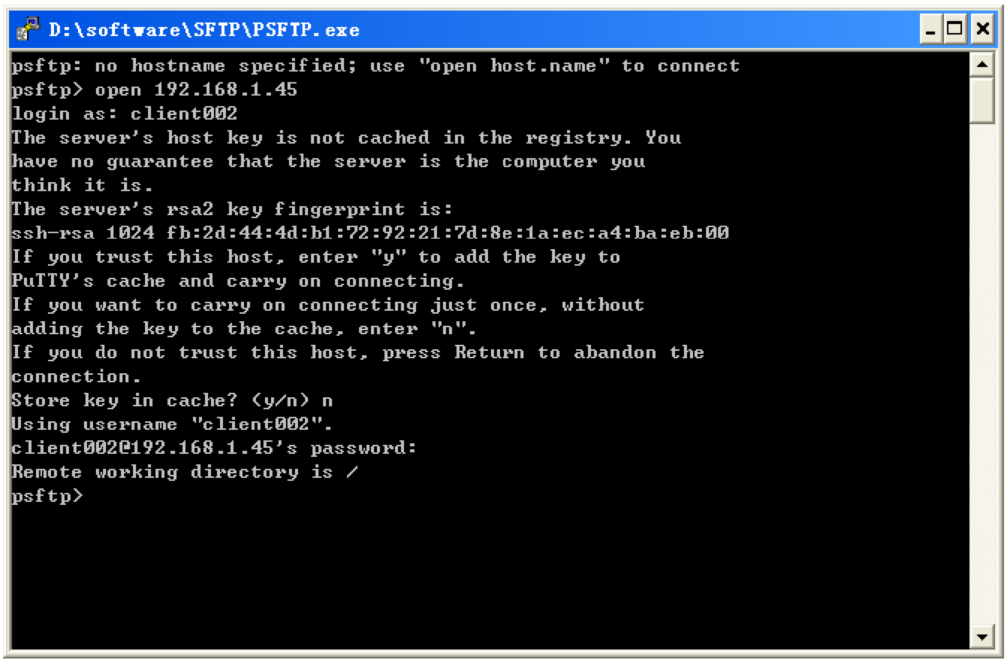
   To establish a connection to the SFTP server:

   a. Run the **psftp.exe** to launch the client interface shown in Figure 14, and enter the following command:

   ```
   open 192.168.1.45
   ```

   b. Enter username **client002** and password **aabbcc** to log in to the SFTP server.

Figure 14 SFTP client interface



```
psftp: no hostname specified; use "open host.name" to connect
psftp> open 192.168.1.45
login as: client002
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 1024 fb:2d:44:4d:b1:72:92:21:7d:8e:1a:ec:a4:ba:eb:00
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) n
Using username "client002".
client002@192.168.1.45's password:
Remote working directory is /
psftp>
```

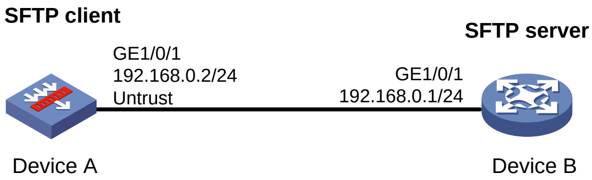
Example: Configuring the device as an SFTP client (publickey authentication)

Network configuration

As shown in Figure 15:

- Device B acts as the SFTP server, and it uses publickey authentication and the RSA public key algorithm.
- Device A acts as the SFTP client. After the user on Device A logs in to Device B through SFTP, the user can perform file management and transfer operations on Device B as a network administrator.

Figure 15 Network diagram



Procedure

In the server configuration, the client's host public key is required. Generate RSA key pairs on the client before configuring the SFTP server.

1. Configure Device A:
   a. Assign IP addresses to interfaces:
   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.2 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

   c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send pacekts to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name sshlocalin
[DeviceA-security-policy-ip-1-sshlocalin] source-zone untrust
[DeviceA-security-policy-ip-1-sshlocalin] destination-zone local
[DeviceA-security-policy-ip-1-sshlocalin] source-ip-host 192.168.0.1
[DeviceA-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.0.2
[DeviceA-security-policy-ip-1-sshlocalin] action pass
[DeviceA-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:
   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

3. Configure the SFTP client:
   # Generate RSA key pairs.

```
[DeviceA] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Export the host public key to a public key file named **pubkey**.

```
[DeviceA] public-key local export rsa ssh2 pubkey
[DeviceA] quit
```

# Transmit the public key file **pubkey** to the server through FTP or TFTP. (Details not shown.)

4. Configure the SFTP server:

# Generate RSA key pairs.

```
<DeviceB> system-view
[DeviceB] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[DeviceB] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[DeviceB] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Enable the SFTP server.

```
[DeviceB] sftp server enable
```

# Import the peer public key from the public key file **pubkey**, and name it **devicekey**.

```
[DeviceB] public-key peer devicekey import sshkey pubkey
```

# Create an SSH user named **client001**. Specify the service type as **sftp** and the authentication method as **publickey** for the user. Assign the public key **devicekey** to the user.

```
[DeviceB] ssh user client001 service-type sftp authentication-type publickey assign
publickey devicekey
```

# Create a local device management user named **client001**.

```
[DeviceB] local-user client001 class manage
```

# Authorize local user **client001** to use the **SSH** service.

```
[DeviceB-luser-manage-client001] service-type ssh
```

# Assign the **network-admin** user role and working directory **flash:/** to local user **client001**.

```
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
work-directory flash:/
[DeviceB-luser-manage-client001] quit
```

5. Establish a connection between the SFTP client and the SFTP server:

# Establish a connection to the SFTP server and enter SFTP client view.

```
<DeviceA> sftp 192.168.0.1 identity-key rsa
Username: client001
Press CTRL+C to abort.
Connecting to 192.168.0.1 port 22.
The server is not authenticated. Continue? [Y/N]:y
Do you want to save the server public key? [Y/N]:n
sftp>
```

# Display files under the current directory of the server, delete file **z**, and verify the result.

```
sftp> dir -l
-rwxrwxrwx   1 noone    nogroup      1759 Aug 23 06:52 config.cfg
-rwxrwxrwx   1 noone    nogroup       225 Aug 24 08:01 pubkey2
-rwxrwxrwx   1 noone    nogroup       283 Aug 24 07:39 pubkey
drwxrwxrwx   1 noone    nogroup         0 Sep 01 06:22 new
-rwxrwxrwx   1 noone    nogroup       225 Sep 01 06:55 pub
-rwxrwxrwx   1 noone    nogroup         0 Sep 01 08:00 z
sftp> delete z
Removing /z
sftp> dir -l
-rwxrwxrwx   1 noone    nogroup      1759 Aug 23 06:52 config.cfg
-rwxrwxrwx   1 noone    nogroup       225 Aug 24 08:01 pubkey2
-rwxrwxrwx   1 noone    nogroup       283 Aug 24 07:39 pubkey
drwxrwxrwx   1 noone    nogroup         0 Sep 01 06:22 new
-rwxrwxrwx   1 noone    nogroup       225 Sep 01 06:55 pub
```

# Add a directory named **new1** and verify the result.

```
sftp> mkdir new1
sftp> dir -l
-rwxrwxrwx   1 noone    nogroup      1759 Aug 23 06:52 config.cfg
-rwxrwxrwx   1 noone    nogroup       225 Aug 24 08:01 pubkey2
-rwxrwxrwx   1 noone    nogroup       283 Aug 24 07:39 pubkey
drwxrwxrwx   1 noone    nogroup         0 Sep 01 06:22 new
-rwxrwxrwx   1 noone    nogroup       225 Sep 01 06:55 pub
drwxrwxrwx   1 noone    nogroup         0 Sep 02 06:30 new1
```

# Change the name of directory **new1** to **new2** and verify the result.

```
sftp> rename new1 new2
sftp> dir
-rwxrwxrwx   1 noone    nogroup      1759 Aug 23 06:52 config.cfg
-rwxrwxrwx   1 noone    nogroup       225 Aug 24 08:01 pubkey2
-rwxrwxrwx   1 noone    nogroup       283 Aug 24 07:39 pubkey
drwxrwxrwx   1 noone    nogroup         0 Sep 01 06:22 new
-rwxrwxrwx   1 noone    nogroup       225 Sep 01 06:55 pub
drwxrwxrwx   1 noone    nogroup         0 Sep 02 06:33 new2
```

# Download file **pubkey2** from the server and save it as a local file named **public**.

```
sftp> get pubkey2 public
Fetching / pubkey2 to public
/pubkey2                                  100% 225     1.4KB/s   00:00
```

# Upload a local file **pu** to the server, save it as **puk**, and verify the result.

```
sftp> put pu puk
Uploading pu to / puk
sftp> dir
-rwxrwxrwx   1 noone    nogroup      1759 Aug 23 06:52 config.cfg
-rwxrwxrwx   1 noone    nogroup       225 Aug 24 08:01 pubkey2
-rwxrwxrwx   1 noone    nogroup       283 Aug 24 07:39 pubkey
drwxrwxrwx   1 noone    nogroup         0 Sep 01 06:22 new
drwxrwxrwx   1 noone    nogroup         0 Sep 02 06:33 new2
-rwxrwxrwx   1 noone    nogroup       283 Sep 02 06:35 pub
-rwxrwxrwx   1 noone    nogroup       283 Sep 02 06:36 puk
sftp>
```

# Exit SFTP client view.

```
sftp> quit
<DeviceA>
```

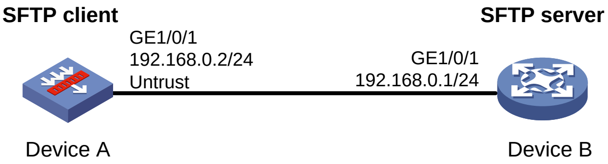
Example: Configuring SFTP based on 192-bit Suite B algorithms

Network configuration

As shown in Figure 16:

- Device B acts as the SFTP Suite B server (SSH2), and it uses publickey authentication to authenticate the SFTP client.
- Device A acts as an SFTP Suite B client (SSH2). After the user on Device A logs in to Device B based on the SFTP Suite B client software, the user can manage and transfer files on Device B as an administrator.

Figure 16 Network diagram



Procedure

1. Generate the client's certificate and the server's certificate. (Details not shown.)

   You must first configure the certificates of the server and the client because they are required for identity authentication between the two parties.

   In this example, the server's certificate file is **ssh-server-ecdsa384.p12** and the client's certificate file is **ssh-client-ecdsa384.p12**.

2. Configure Device A:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   ```

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.2 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send pacekts to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip-1-sshlocalin] source-zone untrust
[DeviceA-security-policy-ip-1-sshlocalin] destination-zone local
[DeviceA-security-policy-ip-1-sshlocalin] source-ip-host 192.168.0.1
[DeviceA-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.0.2
[DeviceA-security-policy-ip-1-sshlocalin] action pass
[DeviceA-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip] quit
```

3. Configure Device B:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

4. Configure the SFTP client:

You can modify the pkix version of the client software OpenSSH to support Suite B. This example uses an NSFOCUS device as an SFTP client.

# Upload the server's certificate file **ssh-server-ecdsa384.p12** and the client's certificate file **ssh-client-ecdsa384.p12** to the SFTP client through FTP or TFTP. (Details not shown.)

# Create a PKI domain named **server384** for verifying the server's certificate and enter its view.

```
<DeviceA> system-view
[DeviceA] pki domain server384
```

# Disable CRL checking.

```
[DeviceA-pki-domain-server384] undo crl check enable
[DeviceA-pki-domain-server384] quit
```

# Import local certificate file **ssh-server-ecdsa384.p12** to PKI domain **server384**.

```
[DeviceA] pki import domain server384 p12 local filename ssh-server-ecdsa384.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: server384]:
```

# Display information about local certificates in PKI domain **server384**.

```
[DeviceA] display pki certificate domain server384 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: ecdsa-with-SHA384
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 20 10:08:41 2015 GMT
            Not After : Aug 19 10:08:41 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=ssh server
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (384 bit)
                pub:
                    04:4a:33:e5:99:8d:49:45:a7:a3:24:7b:32:6a:ed:
                    b6:36:e1:4d:cc:8c:05:22:f4:3a:7c:5d:b7:be:d1:
                    e6:9e:f0:ce:95:39:ca:fd:a0:86:cd:54:ab:49:60:
                    10:be:67:9f:90:3a:18:e2:7d:d9:5f:72:27:09:e7:
                    bf:7e:64:0a:59:bb:b3:7d:ae:88:14:94:45:b9:34:
                    d2:f3:93:e1:ba:b4:50:15:eb:e5:45:24:31:10:c7:
                    07:01:f9:dc:a5:6f:81
                ASN1 OID: secp384r1
                NIST CURVE: P-384
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                10:16:64:2C:DA:C1:D1:29:CD:C0:74:40:A9:70:BD:62:8A:BB:F4:D5
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA384
         30:65:02:31:00:80:50:7a:4f:c5:cd:6a:c3:57:13:7f:e9:da:
         c1:72:7f:45:30:17:c2:a7:d3:ec:73:3d:5f:4d:e3:96:f6:a3:
         33:fb:e4:b9:ff:47:f1:af:9d:e3:03:d2:24:53:40:09:5b:02:
         30:45:d1:bf:51:fd:da:22:11:90:03:f9:d4:05:ec:d6:7c:41:
         fc:9d:a1:fd:5b:8c:73:f8:b6:4c:c3:41:f7:c6:7f:2f:05:2d:
         37:f8:52:52:26:99:28:97:ac:6e:f9:c7:01
```

# Create a PKI domain named **client384** for the client's certificate and enter its view.

```
[DeviceA] pki domain client384
```

# Disable CRL checking.

```
[DeviceA-pki-domain-client384] undo crl check enable
[DeviceA-pki-domain-client384] quit
```

# Import local certificate file **ssh-client-ecdsa384.p12** to PKI domain **client384**.

```
[DeviceA] pki import domain client384 p12 local filename ssh-client-ecdsa384.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: client384]:
```

# Display information about local certificates in PKI domain **client384**.

```
[DeviceA]display pki certificate domain client384 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: ecdsa-with-SHA384
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 20 10:10:59 2015 GMT
            Not After : Aug 19 10:10:59 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=ssh client
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (384 bit)
                pub:
                    04:85:7c:8b:f4:7a:36:bf:74:f6:7c:72:f9:08:69:
                    d0:b9:ac:89:98:17:c9:fc:89:94:43:da:9a:a6:89:
                    41:d3:72:24:9b:9a:29:a8:d1:ba:b4:e5:77:ba:fc:
                    df:ae:c6:dd:46:72:ab:bc:d1:7f:18:7d:54:88:f6:
                    b4:06:54:7e:e7:4d:49:b4:07:dc:30:54:4b:b6:5b:
                    01:10:51:6b:0c:6d:a3:b1:4b:c9:d9:6c:d6:be:13:
                    91:70:31:2a:92:00:76
                ASN1 OID: secp384r1
                NIST CURVE: P-384
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                BD:5F:8E:4F:7B:FE:74:03:5A:D1:94:DB:CA:A7:82:D6:F7:78:A1:B0
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA384
         30:66:02:31:00:d2:06:fa:2c:0b:0d:f0:81:90:01:c3:3d:bf:
         97:b3:79:d8:25:a0:e2:0e:ed:00:c9:48:3e:c9:71:43:c9:b4:
         2a:a6:0a:27:80:9e:d4:0f:f2:db:db:5b:40:b1:a9:0a:e4:02:
         31:00:ee:00:e1:07:c0:2f:12:3f:88:ea:fe:19:05:ef:56:ca:
         33:71:75:5e:11:c9:a6:51:4b:3e:7c:eb:2a:4d:87:2b:71:7c:
         30:64:fe:14:ce:06:d5:0a:e2:cf:9a:69:19:ff
[DeviceA] quit
```

5. Configure the SFTP server:

# Upload the server's certificate file **ssh-server-ecdsa384.p12** and the client's certificate file **ssh-client-ecdsa384.p12** to the SFTP server through FTP or TFTP. (Details not shown.)

# Create a PKI domain named **client384** for verifying the client's certificate and import the file of the client's certificate to this domain. (Details not shown.)

# Create a PKI domain named **server384** for the server's certificate and import the file of the server's certificate to this domain. (Details not shown.)

# Specify Suite B algorithms for algorithm negotiation.

```
[DeviceB] ssh2 algorithm key-exchange ecdh-sha2-nistp384
[DeviceB] ssh2 algorithm cipher aes256-gcm
[DeviceB] ssh2 algorithm public-key x509v3-ecdsa-sha2-nistp384
```

# Specify **server384** as the PKI domain of the server's certificate.

```
[DeviceB] ssh server pki-domain server384
```

# Enable the SFTP server.

```
[DeviceB] sftp server enable
```

# Set the authentication mode to AAA for user lines.

```
[DeviceB] line vty 0 63
[DeviceB-line-vty0-63] authentication-mode scheme
[DeviceB-line-vty0-63] quit
```

# Create a local device management user named **client001**. Authorize the user to use the **SSH** service and assign the **network-admin** user role to the user.

```
[DeviceB] local-user client001 class manage
[DeviceB-luser-manage-client001] service-type ssh
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the **publickey** authentication method for the user and specify **client384** as the PKI domain for verifying the client's certificate.

```
[DeviceB] ssh user client001 service-type sftp authentication-type publickey assign
pki-domain client384
```

6. Establish an SFTP connection to the SFTP server based on the 192-bit Suite B algorithms:

# Establish an SFTP connection to the server at 192.168.0.1.

```
<DeviceA> sftp 192.168.0.1 suite-b 192-bit pki-domain client384 server-pki-domain
server384
Username: client001
Press CTRL+C to abort.
Connecting to 192.168.0.1 port 22.
sftp>
```

SCP configuration examples

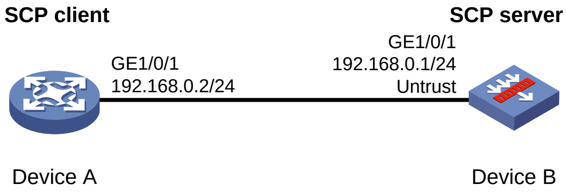
Example: Configuring SCP with password authentication

Network configuration

As shown in Figure 17:

- Device B acts as the SCP server and uses password authentication to authenticate the SCP client. The client's username and password are saved on Device B.
- Device A acts as the SCP client. After the user on Device A logs in to Device B through SCP, the user can transfer files between devices as a network administrator.

Figure 17 Network diagram



Procedure

1. Configure Device A:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.2 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure Device B:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

   c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send pacekts to Device B.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name sshlocalin
[DeviceB-security-policy-ip-1-sshlocalin] source-zone untrust
[DeviceB-security-policy-ip-1-sshlocalin] destination-zone local
[DeviceB-security-policy-ip-1-sshlocalin] source-ip-host 192.168.0.2
[DeviceB-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.0.1
[DeviceB-security-policy-ip-1-sshlocalin] action pass
[DeviceB-security-policy-ip-1-sshlocalin] quit
[DeviceA-security-policy-ip] quit
```

3. Configure the SCP server:

   # Generate RSA key pairs.

```
<DeviceB> system-view
[DeviceB] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
```

```
Input the modulus length [default = 1024]:
Generating Keys...
..
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[DeviceB] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[DeviceB] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Configure the SCP server.

```
[DeviceB] scp server enable
[DeviceB] local-user client001 class manage
[DeviceB-luser-manage-client001] password simple aabbcc
[DeviceB-luser-manage-client001] service-type ssh
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the service type as **scp** and the authentication method as **password** for the user.

```
[DeviceB] ssh user client001 service-type scp authentication-type password
```

4. Connect to the SCP server, download file **remote.bin** from the server, and save it as a local file named **local.bin**.

```
<DeviceA> scp 192.168.0.1 get remote.bin local.bin
Username: client001
Press CTRL+C to abort.
Connecting to 192.168.0.1 port 22.
The server is not authenticated. Continue? [Y/N]:y
Do you want to save the server public key? [Y/N]:n
client001@192.168.0.1’s password:
remote.bin                                       100% 2875     2.8KB/s   00:00
```

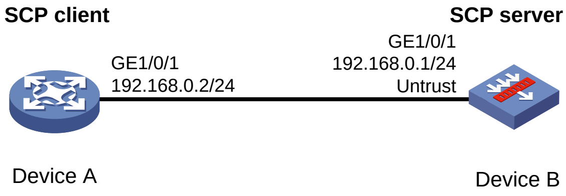
Example: Configuring SCP based on Suite B algorithms

Network configuration

As shown in Figure 18:

- Device B acts as the SCP Suite B server (SSH2), and it uses publickey authentication to authenticate the SCP client.
- Device A acts as an SCP Suite B client (SSH2). After the user on Device A logs in to Device B through SCP based on the SCP Suite B client software, the user can transfer files between devices as a network administrator.

Figure 18 Network diagram



Procedure

1. Generate the client's certificates and the server's certificates. (Details not shown.)

   You must first configure the certificates of the server and the client because they are required for identity authentication between the two parties.

   In this example, the server's certificate files are **ssh-server-ecdsa256.p12** and **ssh-server-ecdsa384.p12**. The client's certificate files are **ssh-client-ecdsa256.p12** and **ssh-client-ecdsa384.p12**.

2. Configure Device A:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 192.168.0.2 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

3. Configure Device B:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name Untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   ```

   c. In the IPv4 security policy, configure a rule to permit traffic between the **Untrust** and **Local** security zones, so Device A can send pacekts to Device B.

   ```
   [DeviceB] security-policy ip
   [DeviceB-security-policy-ip-1-sshlocalin] source-zone untrust
   [DeviceB-security-policy-ip-1-sshlocalin] destination-zone local
   [DeviceB-security-policy-ip-1-sshlocalin] source-ip-host 192.168.0.2
   [DeviceB-security-policy-ip-1-sshlocalin] destination-ip-host 192.168.0.1
   [DeviceB-security-policy-ip-1-sshlocalin] action pass
   [DeviceB-security-policy-ip-1-sshlocalin] quit
   [DeviceA-security-policy-ip] quit
   ```

4. Configure the SCP client:

   You can modify the pkix version of the client software OpenSSH to support Suite B. This example uses an NSFOCUS device as an SCP client.

# Upload the server's certificate files (**ssh-server-ecdsa256.p12** and **ssh-server-ecdsa384.p12**) and the client's certificate files (**ssh-client-ecdsa256.p12** and **ssh-client-ecdsa384.p12**) to the SCP client through FTP or TFTP. (Details not shown.)

# Create a PKI domain named **server256** for verifying the server's certificate **ecdsa256** and enter its view.

```
[DeviceA] pki domain server256
```

# Disable CRL checking.

```
[DeviceA-pki-domain-server256] undo crl check enable
[DeviceA-pki-domain-server256] quit
```

# Import local certificate file **ssh-server-ecdsa256.p12** to PKI domain **server256**.

```
[DeviceA] pki import domain server256 p12 local filename ssh-server-ecdsa256.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: server256]:
```

# Display information about local certificates in PKI domain **server256**.

```
[DeviceA] display pki certificate domain server256 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 3 (0x3)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 21 08:39:51 2015 GMT
            Not After : Aug 20 08:39:51 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=SSH Server secp256
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:a2:b4:b4:66:1e:3b:d5:50:50:0e:55:19:8d:52:
                    6d:47:8c:3d:3d:96:75:88:2f:9a:ba:a2:a7:f9:ef:
                    0a:a9:20:b7:b6:6a:90:0e:f8:c6:de:15:a2:23:81:
                    3c:9e:a2:b7:83:87:b9:ad:28:c8:2a:5e:58:11:8e:
                    c7:61:4a:52:51
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                08:C1:F1:AA:97:45:19:6A:DA:4A:F2:87:A1:1A:E8:30:BD:31:30:D7
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA256
```

```
         30:65:02:31:00:a9:16:e9:c1:76:f0:32:fc:4b:f9:8f:b6:7f:
         31:a0:9f:de:a7:cc:33:29:27:2c:71:2e:f9:0d:74:cb:25:c9:
         00:d2:52:18:7f:58:3f:cc:7e:8b:d3:42:65:00:cb:63:f8:02:
         30:01:a2:f6:a1:51:04:1c:61:78:f6:6b:7e:f9:f9:42:8d:7c:
         a7:bb:47:7c:2a:85:67:0d:81:12:0b:02:98:bc:06:1f:c1:3c:
         9b:c2:1b:4c:44:38:5a:14:b2:48:63:02:2b
```

# Create a PKI domain named **client256** for the client's certificate **ecdsa256** and enter its view.

```
[DeviceA] pki domain client256
```

# Disable CRL checking.

```
[DeviceA-pki-domain-client256] undo crl check enable
[DeviceA-pki-domain-client256] quit
```

# Import local certificate file **ssh-client-ecdsa256.p12** to PKI domain **client256**.

```
[DeviceA] pki import domain client256 p12 local filename ssh-client-ecdsa256.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: client256]:
```

# Display information about local certificates in PKI domain **client256**.

```
[DeviceA] display pki certificate domain client256 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 4 (0x4)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 21 08:41:09 2015 GMT
            Not After : Aug 20 08:41:09 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=SSH Client secp256
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:da:e2:26:45:87:7a:63:20:e7:ca:7f:82:19:f5:
                    96:88:3e:25:46:f8:2f:9a:4c:70:61:35:db:e4:39:
                    b8:38:c4:60:4a:65:28:49:14:32:3c:cc:6d:cd:34:
                    29:83:84:74:a7:2d:0e:75:1c:c2:52:58:1e:22:16:
                    12:d0:b4:8a:92
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                1A:61:60:4D:76:40:B8:BA:5D:A1:3C:60:BC:57:98:35:20:79:80:FC
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA256
         30:66:02:31:00:9a:6d:fd:7d:ab:ae:54:9a:81:71:e6:bb:ad:
         5a:2e:dc:1d:b3:8a:bf:ce:ee:71:4e:8f:d9:93:7f:a3:48:a1:
         5c:17:cb:22:fa:8f:b3:e5:76:89:06:9f:96:47:dc:34:87:02:
         31:00:e3:af:2a:8f:d6:8d:1f:3a:2b:ae:2f:97:b3:52:63:b6:
         18:67:70:2c:93:2a:41:c0:e7:fa:93:20:09:4d:f4:bf:d0:11:
         66:0f:48:56:01:1e:c3:be:37:4e:49:19:cf:c6
```

# Create a PKI domain named **server384** for verifying the server's certificate **ecdsa384** and enter its view.

```
[DeviceA] pki domain server384
```

# Disable CRL checking.

```
[DeviceA-pki-domain-server384] undo crl check enable
[DeviceA-pki-domain-server384] quit
```

# Import local certificate file **ssh-server-ecdsa384.p12** to PKI domain **server384**.

```
[DeviceA] pki import domain server384 p12 local filename ssh-server-ecdsa384.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: server384]:
```

# Display information about local certificates in PKI domain **server384**.

```
[DeviceA] display pki certificate domain server384 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: ecdsa-with-SHA384
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 20 10:08:41 2015 GMT
            Not After : Aug 19 10:08:41 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=ssh server
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (384 bit)
                pub:
                    04:4a:33:e5:99:8d:49:45:a7:a3:24:7b:32:6a:ed:
                    b6:36:e1:4d:cc:8c:05:22:f4:3a:7c:5d:b7:be:d1:
                    e6:9e:f0:ce:95:39:ca:fd:a0:86:cd:54:ab:49:60:
                    10:be:67:9f:90:3a:18:e2:7d:d9:5f:72:27:09:e7:
                    bf:7e:64:0a:59:bb:b3:7d:ae:88:14:94:45:b9:34:
                    d2:f3:93:e1:ba:b4:50:15:eb:e5:45:24:31:10:c7:
                    07:01:f9:dc:a5:6f:81
                ASN1 OID: secp384r1
                NIST CURVE: P-384
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                10:16:64:2C:DA:C1:D1:29:CD:C0:74:40:A9:70:BD:62:8A:BB:F4:D5
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA384
         30:65:02:31:00:80:50:7a:4f:c5:cd:6a:c3:57:13:7f:e9:da:
         c1:72:7f:45:30:17:c2:a7:d3:ec:73:3d:5f:4d:e3:96:f6:a3:
         33:fb:e4:b9:ff:47:f1:af:9d:e3:03:d2:24:53:40:09:5b:02:
         30:45:d1:bf:51:fd:da:22:11:90:03:f9:d4:05:ec:d6:7c:41:
         fc:9d:a1:fd:5b:8c:73:f8:b6:4c:c3:41:f7:c6:7f:2f:05:2d:
         37:f8:52:52:26:99:28:97:ac:6e:f9:c7:01
```

# Create a PKI domain named **client384** for the client's certificate **ecdsa384** and enter its view.

```
[DeviceA] pki domain client384
```

# Disable CRL checking.

```
[DeviceA-pki-domain-client384] undo crl check enable
[DeviceA-pki-domain-client384] quit
```

# Import local certificate file **ssh-client-ecdsa384.p12** to PKI domain **client384**.

```
[DeviceA] pki import domain client384 p12 local filename ssh-client-ecdsa384.p12
The system is going to save the key pair. You must specify a key pair name, which is
a case-insensitive string of 1 to 64 characters. Valid characters include a to z, A
to Z, 0 to 9, and hyphens (-).
Please enter the key pair name[default name: client384]:
```

# Display information about local certificates in PKI domain **client384**.

```
[DeviceA] display pki certificate domain client384 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: ecdsa-with-SHA384
        Issuer: C=CN, ST=Beijing, L=Beijing, O=NSFOCUS, OU=Software, CN=SuiteB CA
        Validity
            Not Before: Aug 20 10:10:59 2015 GMT
            Not After : Aug 19 10:10:59 2016 GMT
        Subject: C=CN, ST=Beijing, O=NSFOCUS, OU=Software, CN=ssh client
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (384 bit)
                pub:
                    04:85:7c:8b:f4:7a:36:bf:74:f6:7c:72:f9:08:69:
                    d0:b9:ac:89:98:17:c9:fc:89:94:43:da:9a:a6:89:
                    41:d3:72:24:9b:9a:29:a8:d1:ba:b4:e5:77:ba:fc:
                    df:ae:c6:dd:46:72:ab:bc:d1:7f:18:7d:54:88:f6:
                    b4:06:54:7e:e7:4d:49:b4:07:dc:30:54:4b:b6:5b:
                    01:10:51:6b:0c:6d:a3:b1:4b:c9:d9:6c:d6:be:13:
```

```
                    91:70:31:2a:92:00:76
                ASN1 OID: secp384r1
                NIST CURVE: P-384
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                BD:5F:8E:4F:7B:FE:74:03:5A:D1:94:DB:CA:A7:82:D6:F7:78:A1:B0
            X509v3 Authority Key Identifier:
                keyid:5A:BE:85:49:16:E5:EB:33:80:25:EB:D8:91:50:B4:E6:3E:4F:B8:22

    Signature Algorithm: ecdsa-with-SHA384
         30:66:02:31:00:d2:06:fa:2c:0b:0d:f0:81:90:01:c3:3d:bf:
         97:b3:79:d8:25:a0:e2:0e:ed:00:c9:48:3e:c9:71:43:c9:b4:
         2a:a6:0a:27:80:9e:d4:0f:f2:db:db:5b:40:b1:a9:0a:e4:02:
         31:00:ee:00:e1:07:c0:2f:12:3f:88:ea:fe:19:05:ef:56:ca:
         33:71:75:5e:11:c9:a6:51:4b:3e:7c:eb:2a:4d:87:2b:71:7c:
         30:64:fe:14:ce:06:d5:0a:e2:cf:9a:69:19:ff
[DeviceA] quit
```

5. Configure the SCP server:

   # Upload the server's certificate files (**ssh-server-ecdsa256.p12** and **ssh-server-ecdsa384.p12**) and the client's certificate files (**ssh-client-ecdsa256.p12** and **ssh-client-ecdsa384.p12**) to the SCP server through FTP or TFTP. (Details not shown.)

   # Create a PKI domain named **client256** for verifying the client's certificate **ecdsa256** and import the file of this certificate to this domain. Create a PKI domain named **server256** for the server's certificate **ecdsa256** and import the file of this certificate to this domain. (Details not shown.)

   # Create a PKI domain named **client384** for verifying the client's certificate **ecdsa384** and import the file of this certificate to this domain. Create a PKI domain named **server384** for the server's certificate **ecdsa384** and import the file of this certificate to this domain. (Details not shown.)

   # Specify Suite B algorithms for algorithm negotiation.

   ```
   <DeviceB> system-view
   [DeviceB] ssh2 algorithm key-exchange ecdh-sha2-nistp256 ecdh-sha2-nistp384
   [DeviceB] ssh2 algorithm cipher aes128-gcm aes256-gcm
   [DeviceB] ssh2 algorithm public-key x509v3-ecdsa-sha2-nistp256
   x509v3-ecdsa-sha2-nistp384
   ```

   # Enable the SCP server.

   ```
   [DeviceB] scp server enable
   ```

   # Set the authentication mode to AAA for user lines.

   ```
   [DeviceB] line vty 0 63
   [DeviceB-line-vty0-63] authentication-mode scheme
   [DeviceB-line-vty0-63] quit
   ```

   # Create a local device management user named **client001**. Authorize the user to use the **SSH** service and assign the **network-admin** user role to the user.

   ```
   [DeviceB] local-user client001 class manage
   [DeviceB-luser-manage-client001] service-type ssh
   ```

```
[DeviceB-luser-manage-client001] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client001] quit
```

# Create a local device management user named **client002**. Authorize the user to use the **SSH** service and assign the **network-admin** user role to the user.

```
[DeviceB] local-user client002 class manage
[DeviceB-luser-manage-client002] service-type ssh
[DeviceB-luser-manage-client002] authorization-attribute user-role network-admin
[DeviceB-luser-manage-client002] quit
```

6. Establish an SCP connection to the SCP server:

- Based on the 128-bit Suite B algorithms:

  # Specify **server256** as the PKI domain of the server's certificate.

  ```
  [DeviceB] ssh server pki-domain server256
  ```

  # Create an SSH user **client001**. Specify the authentication method **publickey** for the user and specify **client256** as the PKI domain for verifying the client's certificate.

  ```
  [DeviceB] ssh user client001 service-type scp authentication-type publickey assign
  pki-domain client256
  ```

  # Establish an SCP connection to the SCP server at 192.168.0.1 based on the 128-bit Suite B algorithms.

  ```
  <DeviceA> scp 192.168.0.1 get src.cfg suite-b 128-bit pki-domain client256
  server-pki-domain server256
  Username: client001
  Press CTRL+C to abort.
  Connecting to 192.168.0.1 port 22.
  src.cfg                                       100% 4814     4.7KB/s   00:00
  <DeviceA>
  ```

- Based on the 192-bit Suite B algorithms:

  # Specify **server384** as the PKI domain of the server's certificate.

  ```
  [DeviceB] ssh server pki-domain server384
  ```

  # Create an SSH user **client002**. Specify the **publickey** authentication method for the user and specify **client384** as the PKI domain for verifying the client's certificate.

  ```
  [DeviceB] ssh user client002 service-type scp authentication-type publickey assign
  pki-domain client384
  ```

  # Establish an SCP connection to the SCP server at 192.168.0.1 based on the 192-bit Suite B algorithms.

  ```
  <DeviceA> scp 192.168.0.1 get src.cfg suite-b 192-bit pki-domain client384
  server-pki-domain server384
  Username: client002
  Press CTRL+C to abort.
  Connecting to 192.168.0.1 port 22.
  src.cfg                                       100% 4814     4.7KB/s   00:00
  <DeviceA>
  ```

NETCONF over SSH configuration examples

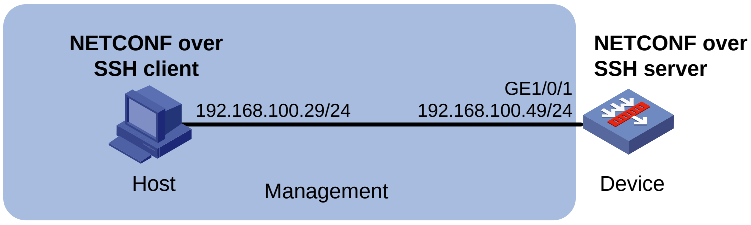
Example: Configuring NETCONF over SSH with password authentication

Network configuration

As shown in Figure 19:

- The device acts as the NETCONF-over-SSH server and uses password authentication to authenticate the client. The client's username and password are saved on the device.
- The host acts as the NETCONF-over-SSH client, using SSH2 client software. After the user on the host logs in to the device through NETCONF over SSH, the user can perform NETCONF operations on the device as a network administrator.

Figure 19 Network diagram



Procedure

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.100.49 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Add interfaces to security zones.

```
[Device] security-zone name management
[Device-security-zone-Management] import interface gigabitethernet 1/0/1
[Device-security-zone-Management] quit
```

# Generate RSA key pairs.

```
[Device] public-key local create rsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
Input the modulus length [default = 1024]:
Generating Keys...
..................
Create the key pair successfully.
```

# Generate a DSA key pair.

```
[Device] public-key local create dsa
The range of public key modulus is (512 ~ 2048).
If the key modulus is greater than 512, it will take a few minutes.
Press CTRL+C to abort.
```

```
Input the modulus length [default = 1024]:
Generating Keys...
.................
Create the key pair successfully.
```

# Generate an ECDSA key pair.

```
[Device] public-key local create ecdsa secp256r1
Generating Keys...
.
Create the key pair successfully.
```

# Enable NETCONF over SSH.

```
[Device] netconf ssh server enable
```

# Set the authentication mode to AAA for user lines.

```
[Device] line vty 0 63
[Device-line-vty0-63] authentication-mode scheme
[Device-line-vty0-63] quit
```

# Create a local device management user named **client001**.

```
[Device] local-user client001 class manage
```

# Set the password to **aabbcc** in plain text for local user **client001**.

```
[Device-luser-manage-client001] password simple aabbcc
```

# Authorize local user **client001** to use the **SSH** service.

```
[Device-luser-manage-client001] service-type ssh
```

# Assign the **network-admin** user role to local user **client001**.

```
[Device-luser-manage-client001] authorization-attribute user-role network-admin
[Device-luser-manage-client001] quit
```

# Create an SSH user named **client001**. Specify the service type as **NETCONF** and the authentication method as **password** for the user.

```
[Device] ssh user client001 service-type netconf authentication-type password
```

Verifying the configuration

1. Launch a client that supports NETCONF over SSH.

   This example uses NetConf Browser 2015 (version 3.1).

2. Select **File** > **Connect…** from the menu.

   The **Connect** page opens, as shown in Figure 20.

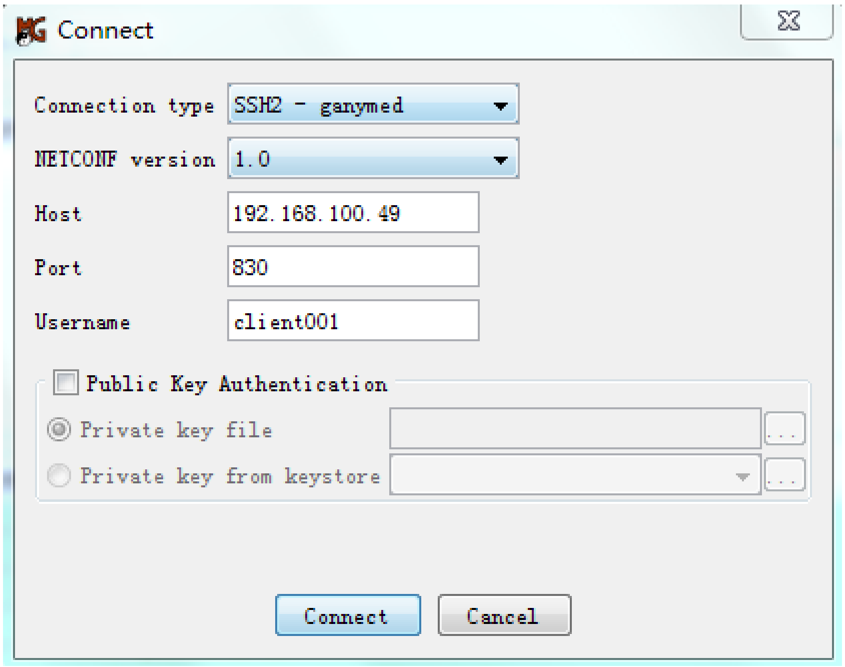
3. Configure connection parameters as follows:
   a. Select a connection type from the **Connection type** list.

      This example uses **SSH2-ganymed**.

   b. Select **1.0** from the **NETCONF version** list.
   c. Enter **192.168.100.49** in the **Host** field.
   d. Enter **830** in the **Port** field.
   e. Enter **client001** in the **Username** field.
   f. Use the default setting for the **Public Key Authentication** area.

4. Click **Connect**.

Figure 20 Connecting to the device



Connection type: SSH2 - ganymed

NETCONF version: 1.0

Host: 192.168.100.49

Port: 830

Username: client001

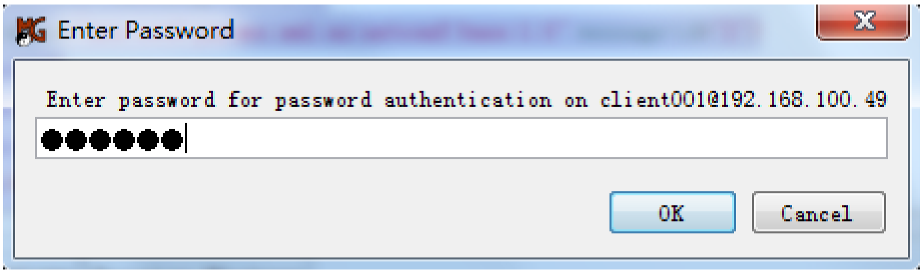
☐ Public Key Authentication

● Private key file

○ Private key from keystore

5. Enter password **aabbcc**, and then click **OK**, as shown in Figure 21.

Figure 21 Entering the password

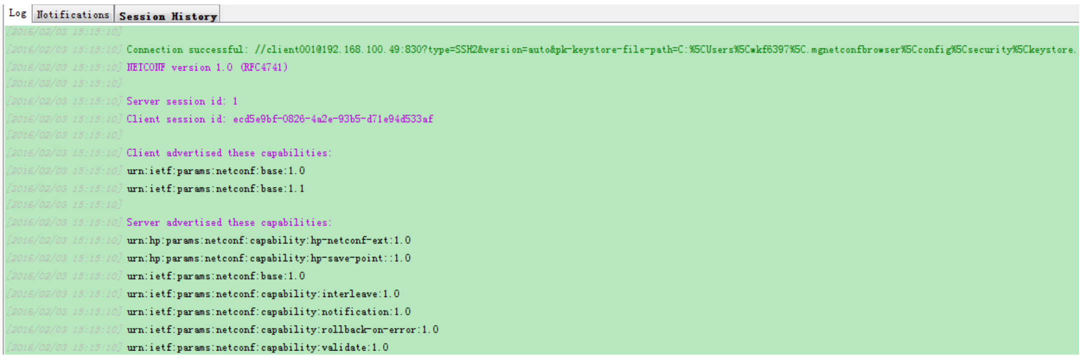


Enter password for password authentication on client001@192.168.100.49

●●●●●●

The NETCONF configuration interface opens when the client successfully establishes an NETCONF-over-SSH connection to the device. The **Log** tab of the interface displays the connection information, as shown in Figure 22.

Figure 22 Logging in to the device



```
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Connection successful: //client001@192.168.100.49:830?type=SSH2&version=auto&pk-keystore-file-path=C:%5CUsers%5Cwkf6397%5C.mgnetconfbrowser%5Cconfig%5Csecurity%5Ckeystore.
[2016/02/03 15:15:10] NETCONF version 1.0 (RFC4741)
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Server session id: 1
[2016/02/03 15:15:10] Client session id: ecd5e9bf-0826-4a2e-93b5-d71e94d533af
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Client advertised these capabilities:
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.1
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Server advertised these capabilities:
[2016/02/03 15:15:10] urn:hp:params:netconf:capability:hp-netconf-ext:1.0
[2016/02/03 15:15:10] urn:hp:params:netconf:capability:hp-save-point::1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:interleave:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:notification:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:rollback-on-error:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:validate:1.0
```

6. Verify that you have obtained the permissions of the **network-admin** user role:

   In the **Command XML** area of the NETCONF configuration interface, enter **<get-sessions/>**, and then click **Send**.

The following message is displayed in the **Output XML** area.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
  <get-sessions>
    <Session>
      <SessionID>1</SessionID>
      <Line>vty1</Line>
      <UserName>client001</UserName>
      <Since>2016-02-03T15:05:30</Since>
      <LockHeld>false</LockHeld>
    </Session>
  </get-sessions>
</rpc-reply>
```

Verifying the configuration

1. Launch a client that supports NETCONF over SSH.

   This example uses NetConf Browser 2015 (version 3.1).

2. Select **File** > **Connect…** from the menu.

   The **Connect** page appears, as shown in Figure 23.

3. Configure connection parameters as follows:

   a. Select a connection type from the **Connection type** list.

      This example uses **SSH2-ganymed**.

   b. Select **1.0** from the **NETCONF version** list.

   c. Enter **192.168.100.49** in the **Host** field.

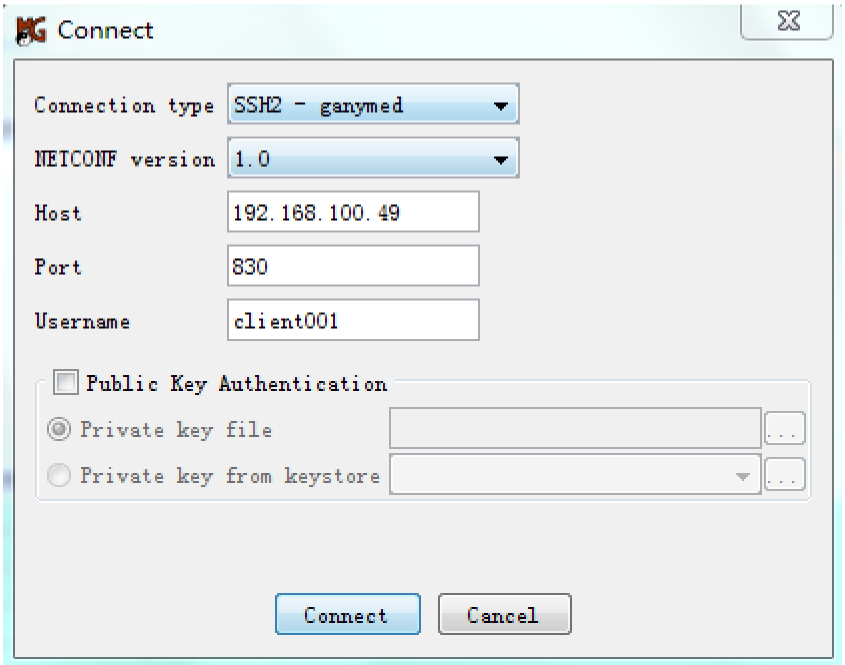
   d. Enter **830** in the **Port** field.

   e. Enter **client001** in the **Username** field.

   f. Use the default setting for the **Public Key Authentication** area.

4. Click **Connect**.

Figure 23 Connecting to the device



Connect

Connection type: SSH2 - ganymed

NETCONF version: 1.0

Host: 192.168.100.49

Port: 830

Username: client001

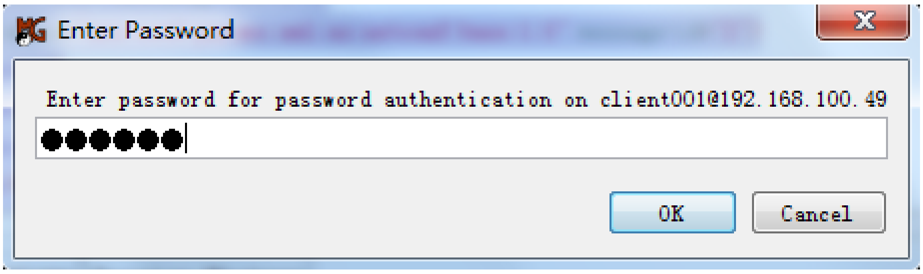
□ Public Key Authentication

● Private key file

○ Private key from keystore

5. Enter password **123456TESTplat&!**, and then click **OK**, as shown in Figure 24.

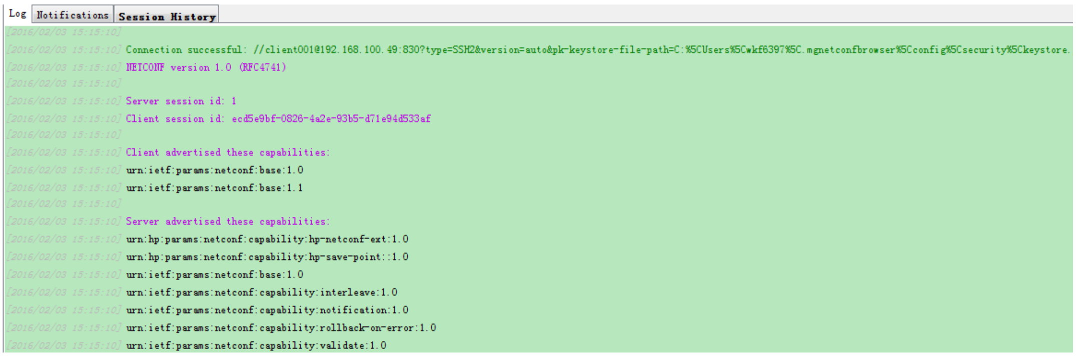
Figure 24 Entering the password



Enter Password
Enter password for password authentication on client001@192.168.100.49
●●●●●●
OK
Cancel

The NETCONF configuration interface appears when the client successfully establishes an NETCONF-over-SSH connection to the device. The **Log** tab of the interface displays the connection information, as shown in Figure 25.

Figure 25 Logging in to the device



```
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Connection successful: //client001@192.168.100.49:830?type=SSH2&version=auto&pk-keystore-file-path=C:%5CUsers%5Cwkf6397%5C.mgnetconfbrowser%5Cconfig%5Csecurity%5Ckeystore.
[2016/02/03 15:15:10] NETCONF version 1.0 (RFC4741)
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Server session id: 1
[2016/02/03 15:15:10] Client session id: ecd5e9bf-0826-4a2e-93b5-d71e94d533af
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Client advertised these capabilities:
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.1
[2016/02/03 15:15:10]
[2016/02/03 15:15:10] Server advertised these capabilities:
[2016/02/03 15:15:10] urn:hp:params:netconf:capability:hp-netconf-ext:1.0
[2016/02/03 15:15:10] urn:hp:params:netconf:capability:hp-save-point::1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:base:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:interleave:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:notification:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:rollback-on-error:1.0
[2016/02/03 15:15:10] urn:ietf:params:netconf:capability:validate:1.0
```

6. In the **Command XML** area of the NETCONF configuration interface, enter **<get-sessions/>**, and then click **Send**.

   The following message is displayed in the **Output XML** area.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
  <get-sessions>
    <Session>
      <SessionID>1</SessionID>
      <Line>vty1</Line>
      <UserName>client001</UserName>
      <Since>2016-02-03T15:05:30</Since>
      <LockHeld>false</LockHeld>
    </Session>
  </get-sessions>
</rpc-reply>
```

Contents

Configuring SSL

About SSL

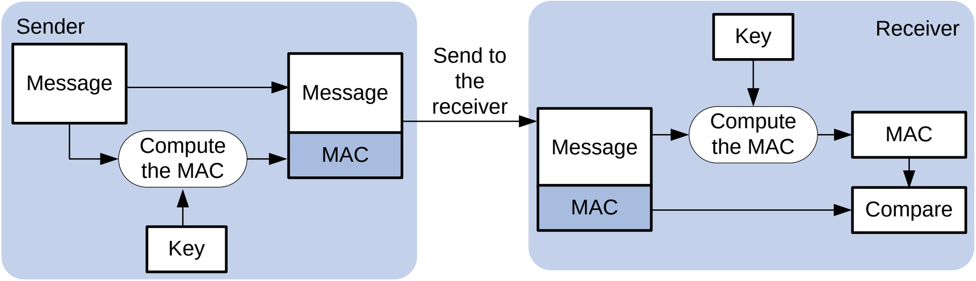
Secure Sockets Layer (SSL) is a cryptographic protocol that provides communication security for TCP-based application layer protocols such as HTTP. SSL has been widely used in applications such as e-business and online banking to provide secure data transmission over the Internet.

SSL security services

SSL provides the following security services:

- **Privacy**—SSL uses a symmetric encryption algorithm to encrypt data. It uses the asymmetric key algorithm of RSA to encrypt the key used by the symmetric encryption algorithm. For more information about RSA, see "Managing public keys."
- **Authentication**—SSL uses certificate-based digital signatures to authenticate the SSL server and client. The SSL server and client obtain digital certificates through PKI. For more information about PKI and digital certificates, see "Configuring PKI."
- **Integrity**—SSL uses the message authentication code (MAC) to verify message integrity. It uses a MAC algorithm and a key to transform a message of any length to a fixed-length message. Any change to the original message will result in a change to the calculated fixed-length message. As shown in Figure 1, the message integrity verification process is as follows:
  - **a.** The sender uses a MAC algorithm and a key to calculate a MAC value for a message. Then, it appends the MAC value to the message and sends the message to the receiver.
  - **b.** The receiver uses the same key and MAC algorithm to calculate a MAC value for the received message, and compares it with the MAC value appended to the message.
  - **c.** If the two MAC values match, the receiver considers the message intact. Otherwise, the receiver considers that the message was tampered with and it discards the message.

Figure 1 MAC algorithm diagram

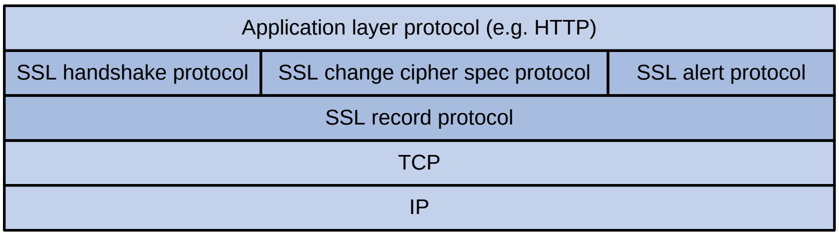


SSL protocol stack

The SSL protocol stack includes the following protocols:

- SSL record protocol at the lower layer.
- SSL handshake protocol, SSL change cipher spec protocol, and SSL alert protocol at the upper layer.

Figure 2 SSL protocol stack


Application layer protocol (e.g. HTTP)
SSL handshake protocol
SSL change cipher spec protocol
SSL alert protocol
SSL record protocol
TCP
IP

The following describes the major functions of SSL protocols:

- **SSL record protocol**—Fragments data received from the upper layer, computes and adds MAC to the data, and encrypts the data.
- **SSL handshake protocol**—Negotiates the cipher suite used for secure communication, authenticates the server and client, and securely exchanges the keys between the server and client. The cipher suite that needs to be negotiated includes the symmetric encryption algorithm, key exchange algorithm, and MAC algorithm.
- **SSL change cipher spec protocol**—Notifies the receiver that subsequent packets are to be protected based on the negotiated cipher suite and key.
- **SSL alert protocol**—Sends alert messages to the receiving party. An alert message contains the alert severity level and a description.

SSL protocol versions

SSL protocol versions include SSL 2.0, SSL 3.0, TLS 1.0 (or SSL 3.1), TLS 1.1, and TLS 1.2. Because SSL 3.0 is known to be insecure, you can disable SSL 3.0 for the SSL server to ensure security.

Restrictions and guidelines: SSL configuration

By default, the SSL server can communicate with clients running all SSL protocol versions. When the server receives an SSL 2.0 Client Hello message from a client, it notifies the client to use a later version for communication.

SSL tasks at a glance

Configuring the SSL server

- Configuring an SSL server policy
- (Optional.) Disabling SSL protocol versions for the SSL server
- (Optional.) Disabling SSL session renegotiation

Configuring the SSL client

Configuring an SSL client policy

Configuring an SSL server policy

About this task

An SSL server policy is a set of SSL parameters used by the device when the device acts as the SSL server. An SSL server policy takes effect only after it is associated with an application such as HTTPS.

Some services (such as SSL VPN, load balancing, and proxy policy) might require using two digital certificates on the server. To meet the requirement, you can use this command to specify two PKI domains in the SSL server policy at a time.

Procedure

1. Enter system view.

   system-view

2. Create an SSL server policy and enter its view.

   ssl server-policy *policy-name*

3. Specify a PKI domain for the SSL server policy.

   pki-domain *domain-name*&<1-2>

   By default, no PKI domain is specified for an SSL server policy.

   If SSL server authentication is required, you must specify a PKI domain and request a local certificate for the SSL server in the domain.

   For information about configuring a PKI domain, see "Configuring PKI."

4. Specify the cipher suites that the SSL server policy supports.

   ciphersuite { **dhe_rsa_aes_128_cbc_sha** | **dhe_rsa_aes_128_cbc_sha256** | **dhe_rsa_aes_256_cbc_sha** | **dhe_rsa_aes_256_cbc_sha256** | **ecc_sm2_sm1_sm3** | **ecc_sm2_sm4_sm3** | **ecdhe_ecdsa_aes_128_cbc_sha256** | **ecdhe_ecdsa_aes_128_gcm_sha256** | **ecdhe_ecdsa_aes_256_cbc_sha384** | **ecdhe_ecdsa_aes_256_gcm_sha384** | **ecdhe_rsa_aes_128_cbc_sha256** | **ecdhe_rsa_aes_128_gcm_sha256** | **ecdhe_rsa_aes_256_cbc_sha384** | **ecdhe_rsa_aes_256_gcm_sha384** | **ecdhe_sm2_sm1_sm3** | **ecdhe_sm2_sm4_sm3** | **exp_rsa_des_cbc_sha** | **rsa_3des_ede_cbc_sha** | **rsa_aes_128_cbc_sha** | **rsa_aes_128_cbc_sha256** | **rsa_aes_128_gcm_sha256** | **rsa_aes_256_cbc_sha** | **rsa_aes_256_cbc_sha256** | **rsa_aes_256_gcm_sha384** | **rsa_des_cbc_sha** | **rsa_sm1_sha** | **rsa_sm1_sm3** | **rsa_sm4_sha** | **rsa_sm4_sm3** | **tls_aes_128_ccm_8_sha256** | **tls_aes_128_ccm_sha256** | **tls_aes_128_gcm_sha256** | **tls_aes_256_gcm_sha384** | **tls_chacha20_poly1305_sha256** } *<1-11>

   By default, an SSL server policy supports cipher suites ECC_SM2_SM1_SM3, ECC_SM2_SM4_SM3, ECDHE_SM2_SM1_SM3, ECDHE_SM2_SM4_SM3, RSA_SM1_SHA, RSA_SM1_SM3, RSA_SM4_SHA, RSA_SM4_SM3, RSA_AES_128_CBC_SHA, RSA_AES_256_CBC_SHA, DHE_RSA_AES_128_CBC_SHA, DHE_RSA_AES_256_CBC_SHA, RSA_AES_128_CBC_SHA256, RSA_AES_256_CBC_SHA256, DHE_RSA_AES_128_CBC_SHA256, DHE_RSA_AES_256_CBC_SHA256, ECDHE_RSA_AES_128_CBC_SHA256, ECDHE_RSA_AES_256_CBC_SHA384, ECDHE_RSA_AES_128_GCM_SHA256, ECDHE_RSA_AES_256_GCM_SHA384, ECDHE_ECDSA_AES_128_CBC_SHA256, ECDHE_ECDSA_AES_256_CBC_SHA384, ECDHE_ECDSA_AES_128_GCM_SHA256, ECDHE_ECDSA_AES_256_GCM_SHA384, RSA_AES_128_GCM_SHA256, RSA_AES_256_GCM_SHA384, TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256, TLS_AES_128_CCM_SHA256, and TLS_AES_128_CCM_8_SHA256.

5. (Optional.) Set the maximum number of sessions that the SSL server can cache and the session cache timeout time.

session { **cachesize** *size* | **timeout** *time* } *

By default, the SSL server can cache a maximum of 500 sessions, and the session cache timeout time is 3600 seconds.

6. Enable mandatory or optional SSL client authentication.

   client-verify { **enable** | **optional** }

   By default, SSL client authentication is disabled. The SSL server does not perform digital certificate-based authentication on SSL clients.

   When authenticating a client by using the digital certificate, the SSL server verifies the certificate chain presented by the client. It also verifies that the certificates in the certificate chain (except the root CA certificate) are not revoked.

7. (Optional.) Enable the SSL server to send the complete certificate chain to the client during SSL negotiation.

   certificate-chain-sending enable

   By default, the SSL server sends the server certificate rather than the complete certificate chain to the client during negotiation.

Configuring an SSL client policy

About this task

An SSL client policy is a set of SSL parameters used by the device when the device acts as the SSL client. The SSL client uses the settings in the client policy to establish a connection to the server. An SSL client policy takes effect only after it is associated with an application such as DDNS. For information about DDNS, see *Layer 3—IP Services Configuration Guide*.

Restrictions and guidelines

As a best practice to enhance system security, do not specify SSL 3.0 for the SSL client policy.

Procedure

1. Enter system view.

   system-view

2. Create an SSL client policy and enter its view.

   ssl client-policy *policy-name*

3. Specify a PKI domain for the SSL client policy.

   pki-domain *domain-name*

   By default, no PKI domain is specified for an SSL client policy.

   If SSL client authentication is required, you must specify a PKI domain and request a local certificate for the SSL client in the PKI domain.

   For information about configuring a PKI domain, see "Configuring PKI."

4. Specify the preferred cipher suite for the SSL client policy.

   prefer-cipher { **dhe_rsa_aes_128_cbc_sha** | **dhe_rsa_aes_128_cbc_sha256** | **dhe_rsa_aes_256_cbc_sha** | **dhe_rsa_aes_256_cbc_sha256** | **ecc_sm2_sm1_sm3** | **ecc_sm2_sm4_sm3** | **ecdhe_ecdsa_aes_128_cbc_sha256** | **ecdhe_ecdsa_aes_128_gcm_sha256** | **ecdhe_ecdsa_aes_256_cbc_sha384** | **ecdhe_ecdsa_aes_256_gcm_sha384** | **ecdhe_rsa_aes_128_cbc_sha256** | **ecdhe_rsa_aes_128_gcm_sha256** | **ecdhe_rsa_aes_256_cbc_sha384** | **ecdhe_rsa_aes_256_gcm_sha384** | **ecdhe_sm2_sm1_sm3** | **ecdhe_sm2_sm4_sm3** | **exp_rsa_des_cbc_sha** | **rsa_3des_ede_cbc_sha** | **rsa_aes_128_cbc_sha** | **rsa_aes_128_cbc_sha256** | **rsa_aes_128_gcm_sha256** | **rsa_aes_256_cbc_sha** | **rsa_aes_256_cbc_sha256** | **rsa_aes_256_gcm_sha384** | **rsa_des_cbc_sha** | **rsa_sm1_sha** | **rsa_sm1_sm3**

| rsa_sm4_sha | rsa_sm4_sm3 | tls_aes_128_ccm_8_sha256 | tls_aes_128_ccm_sha256 | tls_aes_128_gcm_sha256 | tls_aes_256_gcm_sha384 | tls_chacha20_poly1305_sha256 } *<1-11>

By default, the preferred cipher suites of an SSL client policy are **dhe_rsa_aes_256_cbc_sha, rsa_aes_256_cbc_sha**, **dhe_rsa_aes_128_cbc_sha**, and **rsa_aes_128_cbc_sha**.

5. Specify the SSL protocol version for the SSL client policy.

   version { **gm-tls1.1** | **ssl3.0** | **tls1.0** | **tls1.1** | **tls1.2** | **tls1.3** }

   Support for the **gm-tls1.1** keyword depends on the device model. For more information, see the command reference.

   By default, an SSL client policy uses TLS 1.2.

6. Enable the SSL client to authenticate servers through digital certificates.

   server-verify enable

   By default, SSL server authentication is enabled.

Disabling SSL protocol versions for the SSL server

About this task

To enhance security, you can disable the SSL server from using specific SSL protocol versions for session negotiation.

You can disable an SSL protocol version for the SSL server in system view or in SSL server policy view. The SSL server can use an SSL protocol version for session negotiation only when the status of the SSL protocol version in the SSL server policy is **Enabled**. The status of an SSL protocol version in an SSL server policy is determined in the following sequence:

1. Configuration of the **version disable** command in SSL server policy view.
2. Configuration of the **ssl version disable** command in system view.
3. Default setting (**Enabled**).

Make sure the SSL server is allowed to use a minimum of one SSL protocol version for session negotiation.

Restrictions and guidelines

Disabling an SSL protocol version does not affect the availability of earlier SSL protocol versions. For example, if you execute the **ssl version tls1.1 disable** command, TLS 1.1 is disabled but TLS 1.0 is still available for the SSL server.

Procedure

1. Enter system view.

   system-view

2. Disable SSL protocol versions for the SSL server in system view.

   ssl version { **gm-tls1.1** | **ssl3.0** | **tls1.0** | **tls1.1** | **tls1.2** | **tls1.3** } * **disable**

   Support for the **gm-tls1.1** keyword depends on the device model. For more information, see the command reference.

   By default, the SSL server supports TLS 1.1 and TLS 1.2.

3. Enter SSL server policy view.

   ssl server-policy *policy-name*

4. Disable SSL protocol versions in the SSL server policy.

   version { **gm-tls1.1** | **ssl3.0** | **tls1.0** | **tls1.1** | **tls1.2** | **tls1.3** } * **disable**

   Support for the **gm-tls1.1** keyword depends on the device model. For more information, see the command reference.

   By default, an SSL protocol version is enabled in an SSL sever policy unless it is explicitly disabled in system view by using the **ssl version disable** command.

Disabling SSL session renegotiation

About this task

The SSL session renegotiation feature enables the SSL client and server to reuse a previously negotiated SSL session for an abbreviated handshake.

Disabling session renegotiation causes more computational overhead to the system but it can avoid potential risks.

Restrictions and guidelines

Disable SSL session renegotiation only when explicitly required.

Procedure

1. Enter system view.

   system-view

2. Disable SSL session renegotiation.

   ssl renegotiation disable

   By default, SSL session renegotiation is disabled.

Enabling the server-preferred order during cipher suite negotiation

About this task

During SSL connection negotiation, the key exchange algorithm, symmetric encryption algorithm, and MAC algorithm used for message exchange between the SSL server and the SSL client will be determined. By default, the SSL server uses the order of cipher suites presented by the client to negotiate the cipher suite. That is, the SSL server chooses the first cipher suite in the client's list that matches any one of the server's cipher suites. If no match is found, the negotiation fails.

This feature allows you to select the server-preferred order for cipher suite negotiation. That is, the SSL server chooses the first cipher suite in its list that matches any one of the client's cipher suites. If no match is found, the negotiation fails.

The earlier a cipher suite is configured, the higher priority it has during the cipher suite negotiation.

Procedure

1. Enter system view.

   system-view

2. Enter SSL server policy view.

   ssl server-policy *policy-name*

3. Enable the server-preferred order for choosing a cipher suite during the cipher suite negotiation between the SSL server and SSL client.

ciphersuite server-preferred enable

By default, the client-preferred order for choosing a cipher suite during the cipher suite negotiation between the SSL server and SSL client.

Display and maintenance commands for SSL

Execute **display** commands in any view.

Task	Command
Display SSL client policy information.	**display ssl client-policy** [*policy-name*]
Display SSL server policy information.	**display ssl server-policy** [*policy-name*]

Contents

Configuring connection limits

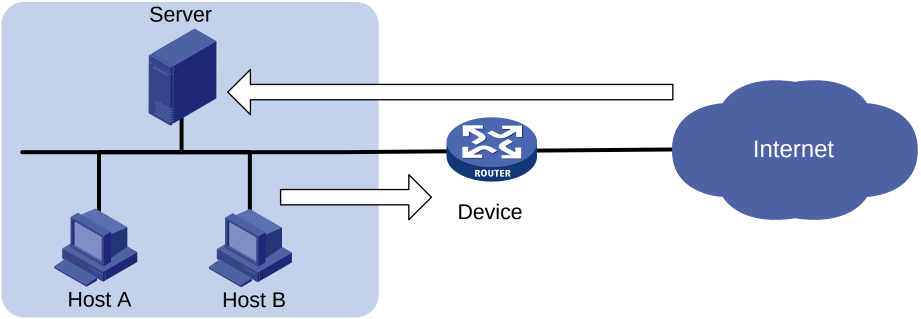
About connection limits

The connection limit feature enables the device to monitor and limit the number of established connections.

As shown in Figure 1, configure the connection limit feature to resolve the following issues:

- If Host B initiates a large number of connections in a short period of time, it might exhaust system resources and cause Host A to be unable to access the Internet.
- If the internal server receives a large number of connection requests in a short period of time, the server cannot process other requests.

Figure 1 Network diagram



Connection limit tasks at a glance

To configure connection limits, perform the following tasks:

1. Creating a connection limit policy
2. Configuring the connection limit policy
   - Configuring an IPv4 connection limit policy
   - Configuring an IPv6 connection limit policy
3. Applying the connection limit policy
   - Applying a connection limit policy to an interface
   - Applying a connection limit policy globally

Creating a connection limit policy

1. Enter system view.
   system-view
2. Create a connection limit policy and enter its view.
   connection-limit { **ipv6-policy** | **policy** } *policy-id*

Configuring the connection limit policy

About connection limit policies

To use a connection limit policy, you need to add limit rules to the policy. Each rule defines a range of connections and the criteria for limiting the connections. Connections in the range will be limited based on the criteria. The criteria include upper/lower connection limit and connection establishment rate limit. When the number of matching connections reaches the upper limit or connection rate limit, the device accepts or rejects new connections depending on the action you configured. If the action is to reject new connections, the device does not accept new connections until the number of connections drops below the lower limit or connection rate limit due to connection aging. The device will send logs when the number of connections exceeds the upper limit or connection rate limit. The device will send logs when the number of connections drops below the lower limit or connection rate limit only if the action is to reject new connections. The connections that do not match any connection limit rules are not limited.

In each connection limit rule, an ACL is used to define the connection range. In addition, the rule also uses the following filtering methods to further limit the connections:

- **per-destination**—Limits user connections by destination IP address.
- **per-service**—Limits user connections by service (transport layer protocol and service port).
- **per-source**—Limits user connections by source IP address.
- **per-dslite-b4**—Limits user connections by the B4 device on a DS-Lite tunnel. For information about DS-Lite tunnels, see *VPN Configuration Guide*.

You can select more than one filtering method, and the selected methods take effect at the same time. For example, if you specify both **per-destination** and **per-service**, the user connections using the same service and destined to the same IP address are limited. If you do not specify any filtering methods in a limit rule, all user connections in the range are limited.

Restrictions and guidelines for connection limit policy configuration

When a connection limit policy is applied, connections on the device match all limit rules in the policy in ascending order of rule IDs. As a best practice, specify a smaller range and more filtering methods in a rule with a smaller ID.

Configuring an IPv4 connection limit policy

1. Enter system view.

   system-view

2. Create an IPv4 connection limit policy and enter its view.

   connection-limit policy *policy-id*

3. Configure a connection limit rule.
   - Configure a connection limit rule based on destination IP address, service, or source IP address.

     limit *limit-id* **acl** { *acl-number* | **name** *acl-name* } [**per-destination** | **per-service** | **per-source**] * { **amount** *max-amount min-amount* | **rate** *rate* } * [**description** *text* | **permit-new**] *

   - Configure a connection limit rule based on DS-Lite tunnel.

limit *limit-id* **acl ipv6** { *acl-number* | **name** *acl-name* } **per-dslite-b4** { **amount** *max-amount min-amount* | **rate** *rate* } * [**description** *text* | **permit-new**] *

4. (Optional.) Configure a description for the connection limit policy.

   description *text*

   By default, an IPv4 connection limit policy does not have a description.

Configuring an IPv6 connection limit policy

1. Enter system view.

   system-view
2. Create an IPv6 connection limit policy and enter its view.

   connection-limit ipv6-policy *policy-id*
3. Configure a connection limit rule.

   limit *limit-id* **acl ipv6** { *acl-number* | **name** *acl-name* } [**per-destination** | **per-service** | **per-source**] * { **amount** *max-amount min-amount* | **rate** *rate* } * [**description** *text* | **permit-new**] *
4. (Optional.) Configure a description for the connection limit policy.

   description *text*

   By default, an IPv6 connection limit policy does not have a description.

Applying the connection limit policy

About connection limit application

To make a connection limit policy take effect, apply it globally or to an interface. The connection limit policy applied to an interface takes effect only on the specified connections on the interface. The connection limit policy applied globally takes effect on all the specified connections on the device.

Different connection limit policies can be applied to individual interfaces as well as globally on the device. In this case, the device matches connections against these policies in the order of the policy on the inbound interface, the global policy, and the policy on the outbound interface. It cannot accept new connections as long as the number of connections reaches the smallest upper connection limit defined by these policies.

Restrictions and guidelines for connection limit application

A connection limit policy or any modification to it takes effect only on new connections. It does not take effect on existing connections.

On an IRF fabric where session synchronization is enabled, connection limit policies applied to a subordinate device do not take effect on sessions switched from the master device.

On a DS-Lite tunnel network, if the AFTR device uses the Endpoint-Independent Mapping-based NAT configuration, you must limit connections from external IPv4 networks to access the internal IPv4 network. To implement B4 device-based connection limits, perform the following tasks:

- Add a rule that has the **per-dslite-b4** keyword to a connection limit policy.
- Apply the policy globally or on the DS-Lite tunnel interface.

Applying a connection limit policy to an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Apply a connection limit policy to the interface.

   connection-limit apply { **ipv6-policy** | **policy** } *policy-id*

   By default, no connection limit policy is applied to an interface.

   Only one IPv4 connection limit policy and one IPv6 connection limit policy can be applied to an interface. A new IPv4 or IPv6 connection limit policy overwrites the old policy.

Applying a connection limit policy globally

1. Enter system view.

   system-view

2. Apply a connection limit policy globally.

   connection-limit apply global { **ipv6-policy** | **policy** } *policy-id*

   By default, no connection limit policy is applied globally.

   Only one IPv4 connection limit policy and one IPv6 connection limit policy can be applied globally. A new IPv4 or IPv6 connection limit policy overwrites the old policy.

Display and maintenance commands for connection limits

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the connection limit policy information.	**display connection-limit** { **ipv6-policy** | **policy** } { **all** | *policy-id* }
Display the connection limit statistics globally or on an interface.	**display connection-limit statistics** { **global** | **interface** *interface-type interface-number* } [**slot** *slot-number*]
Display statistics about IPv4 connections matching connection limit rules globally or on an interface.	**display connection-limit stat-nodes** { **global** | **interface** *interface-type interface-number* } [**slot** *slot-number*] [{ **deny-new** | **permit-new** } | **destination** *destination-ip* | **service-port** *port-number* | **source** *source-ip*] * [**count**] **display connection-limit stat-nodes** { **global** | **interface** *interface-type interface-number* } [**slot** *slot-number*] **dslite-peer** *b4-address* [**count**]
Display statistics about IPv6 connections matching connection limit rules globally or on an interface.	**display connection-limit ipv6-stat-nodes** { **global** | **interface** *interface-type interface-number* } [**slot** *slot-number*]

Task	Command
	[{ **deny-new** | **permit-new** } | **destination** *destination-ip* | **service-port** *port-number* | **source** *source-ip*] * [**count**]
Clear the connection limit statistics globally or on an interface.	**reset connection-limit statistics** { **global** | **interface** *interface-type interface-number* } [**slot** *slot-number*]

Connection limit configuration examples

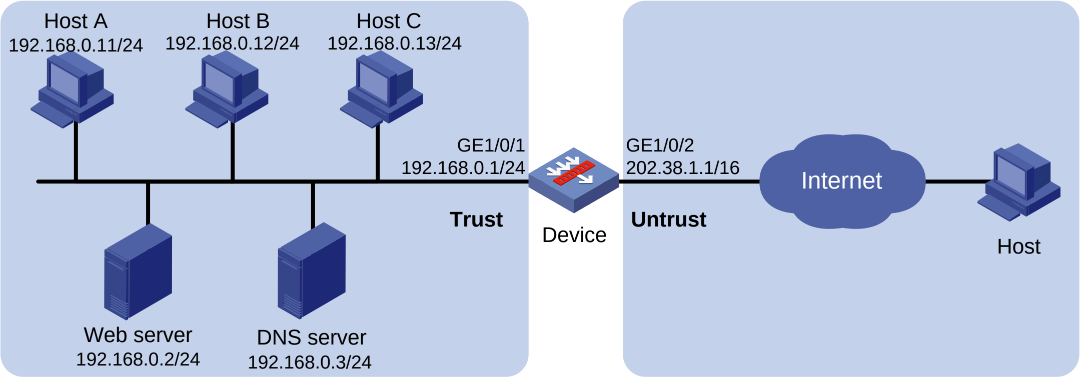
Example: Configuring connection limits

Network configuration

As shown in Figure 2, a company has five public IP addresses: 202.38.1.1/24 to 202.38.1.5/24. The internal network address is 192.168.0.0/16. Configure NAT so that the internal users can access the Internet and external users can access the internal servers. Configure connection limits to meet the following requirements:

- All hosts on segment 192.168.0.0/24 can establish a maximum of 100000 connections to the external network.
- Each host on segment 192.168.0.0/24 can establish a maximum of 100 connections to the external network.
- A maximum of 10000 query requests from DNS clients to the DNS server are allowed at the same time.
- A maximum of 10000 connection requests from Web clients to the Web server are allowed at the same time.

Figure 2 Network diagram



Procedure

The following example only describes how to configure connection limits. For information about NAT configuration and internal server configuration, see *Layer 3—IP Services Configuration Guide*.

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
```

```
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 202.38.1.254.

```
[Device] ip route-static 0.0.0.0 0 202.38.1.254
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a security policy.

   # Configure a rule named **trust-untrust** to allow Host A to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 24
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to allow the external host to access internal servers.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-2-untrust-trust] source-zone untrust
[Device-security-policy-ip-2-untrust-trust] destination-zone trust
[Device-security-policy-ip-2-untrust-trust] destination-ip-subnet 192.168.0.0 24
[Device-security-policy-ip-2-untrust-trust] action pass
[Device-security-policy-ip-2-untrust-trust] quit
[Device-security-policy-ip] quit
```

5. Configure ACLs.

   # Create ACL 3000 to permit packets from all hosts on the internal network.

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip source 192.168.0.0 0.0.0.255
[Device-acl-ipv4-adv-3000] quit
```

# Create ACL 3001 to permit packets to the Web server and the DNS server.

```
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3001] rule permit ip destination 192.168.0.2 0
[Device-acl-ipv4-adv-3001] rule permit ip destination 192.168.0.3 0
[Device-acl-ipv4-adv-3001] quit
```

6. Configure connection limit policies.

   # Create connection limit policy 1.

```
[Device] connection-limit policy 1
```

# Configure connection limit rule 1 to permit a maximum of 100000 connections from all the hosts that match ACL 3000. When the number of connections exceeds 100000, new connections cannot be established until the number drops below 95000.

```
[Device-connlmt-policy-1] limit 1 acl 3000 amount 100000 95000
```

# Configure connection limit rule 2 to permit a maximum of 10000 connections to the servers that match ACL 3001. When the number of connections exceeds 10000, new connections cannot be established until the number drops below 9800.

```
[Device-connlmt-policy-1] limit 2 acl 3001 per-destination amount 10000 9800
[Device-connlmt-policy-1] quit
```

# Create connection limit policy 2.

```
[Device] connection-limit policy 2
```

# Configure connection limit rule 1 to permit a maximum of 100 connections from each host matching ACL 3000. When the number of connections exceeds 100, new connections cannot be established until the number drops below 90.

```
[Device-connlmt-policy-2] limit 1 acl 3000 per-source amount 100 90
[Device-connlmt-policy-2] quit
```

7. Apply connection limit policies.

# Apply connection limit policy 1 globally.

```
[Device] connection-limit apply global policy 1
```

# Apply connection limit policy 2 to inbound interface GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] connection-limit apply policy 2
[Device-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Display information about the connection limit policy.

```
[Device] display connection-limit policy 1
IPv4 connection limit policy 1 has been applied 1 times, and has 2 limit rules.
Limit rule list:
  Policy  Rule    StatType  HiThres  LoThres   rate    PermitNew   ACL
-------------------------------------------------------------------------------
       1     1          --   100000    95000      0           No   3000
             2         Dst    10000     9800      0           No   3001

 Applied list:
     Global
[Device] display connection-limit policy 2
IPv4 connection limit policy 2 has been applied 1 times, and has 1 limit rules.
Limit rule list:
  Policy  Rule    StatType  HiThres  LoThres   rate     PermitNew   ACL
--------------------------------------------------------------------------------
       2     1         Src      100       90      0            No   3000

 Applied list:
     GigabitEthernet1/0/1
```

Troubleshooting connection limits

ACLs in the connection limit rules with overlapping segments

Symptom

A connection limit policy has two rules. Rule 1 sets the upper limit to 10 for the connections from each host on segment 192.168.0.0/24. Rule 2 sets the upper limit to 100 for the connections from 192.168.0.100/24.

```
<Device> system-view
[Device] acl basic 2001
[Device-acl-ipv4-basic-2001] rule permit source 192.168.0.0 0.0.0.255
[Device-acl-ipv4-basic-2001] quit
[Device] acl basic 2002
[Device-acl-ipv4-basic-2002] rule permit source 192.168.0.100 0
[Device-acl-ipv4-basic-2002] quit
[Device] connection-limit policy 1
[Device-connlmt-policy-1] limit 1 acl 2001 per-destination amount 10 5
[Device-connlmt-policy-1] limit 2 acl 2002 per-destination amount 100 10
```

As a result, the host at 192.168.0.100 can only initiate a maximum of 10 connections to the external network.

Solution

To resolve the issue:

1. Rearrange the two connection limit rules by exchanging their rule IDs.
2. If the issue persists, contact NSFOCUS Support.

Contents

Configuring attack detection and prevention

About attack detection and prevention

Attack detection and prevention enables a device to detect attacks by inspecting arriving packets, and to take prevention actions to protect a private network. Prevention actions include logging, packet dropping, blacklisting, and client verification.

Attacks that the device can prevent

This section describes the attacks that the device can detect and prevent.

Single-packet attacks

Single-packet attacks are also known as malformed packet attacks. An attacker typically launches single-packet attacks by using the following methods:

- An attacker sends defective packets to a device, which causes the device to malfunction or crash.
- An attacker sends normal packets to a device, which interrupts connections or probes network topologies.
- An attacker sends a large number of forged packets to a target device, which consumes network bandwidth and causes denial of service (DoS).

Table 1 lists the single-packet attack types that the device can detect and prevent.

Table 1 Types of single-packet attacks

Single-packet attack	Description
ICMP redirect	An attacker sends ICMP redirect messages to modify the victim's routing table. The victim cannot forward packets correctly.
ICMP destination unreachable	An attacker sends ICMP destination unreachable messages to cut off the connections between the victim and its destinations.
ICMP type	A receiver responds to an ICMP packet according to its type. An attacker sends forged ICMP packets of a specific type to affect the packet processing of the victim.
ICMPv6 type	A receiver responds to an ICMPv6 packet according to its type. An attacker sends forged ICMPv6 packets of specific types to affect the packet processing of the victim.
Land	An attacker sends the victim a large number of TCP SYN packets, which contain the victim's IP address as the source and destination IP addresses. This attack exhausts the half-open connection resources on the victim, and locks the victim's system.
Large ICMP packet	An attacker sends large ICMP packets to crash the victim. Large ICMP packets can cause memory allocation error and crash the protocol stack.
Large ICMPv6 packet	An attacker sends large ICMPv6 packets to crash the victim. Large ICMPv6 packets can cause memory allocation error and crash the protocol stack.

Single-packet attack	Description
IP option	An attacker builds IP datagrams with certain option types and sends them to probe the network topology.
IP option abnormal	An attacker sends IP datagrams in which the IP options are abnormal. This attack intends to probe the network topology. The target system will break down if it is incapable of processing error packets.
IP fragment	An attacker sends the victim an IP datagram with an offset no larger than 5, which causes the victim to malfunction or crash.
IP impossible packet	An attacker sends IP packets whose source IP address is the same as the destination IP address, which causes the victim to malfunction.
Tiny fragment	An attacker makes the fragment size small enough to force Layer 4 header fields into the second fragment. These fragments can pass the packet filtering because they do not hit any match.
Smurf	An attacker sends an ICMP echo request to target networks. In these requests, the destination IP address is a network or broadcast address of a Class A, B, or C subnet, and the source IP address is the victim's IP address. Every receiver on the target networks will send an ICMP echo reply to the victim. The victim will be flooded with replies, and will be unable to provide services. Network congestion might occur.
TCP flag	An attacker sends packets with defective TCP flags to probe the operating system of the target host. Different operating systems process unconventional TCP flags differently. The target system will break down if it processes this type of packets incorrectly.
Traceroute	An attacker uses traceroute tools to probe the topology of the victim network.
WinNuke	An attacker sends Out-Of-Band (OOB) data to the TCP port 139 (NetBIOS) on the victim that runs Windows system. The malicious packets contain an illegal Urgent Pointer, which causes the victim's operating system to crash.
UDP bomb	An attacker sends a malformed UDP packet. The length value in the IP header is larger than the IP header length plus the length value in the UDP header. When the target system processes the packet, a buffer overflow can occur, which causes a system crash.
UDP Snork	An attacker sends a UDP packet with destination port 135 (the Microsoft location service) and source port 135, 7, or 19. This attack causes an NT system to exhaust its CPU.
UDP Fraggle	An attacker sends a large number of packets with source UDP port 7 and destination UDP port 19 (UDP chargen port) to a network. These packets use the victim's IP address as the source IP address. Replies will flood the victim, resulting in DoS.
Teardrop	An attacker sends a stream of overlapping fragments. The victim will crash when it tries to reassemble the overlapping fragments.
Ping of death	An attacker sends the victim an ICMP echo request larger than 65535 bytes that violates the IP protocol. When the victim reassembles the packet, a buffer overflow can occur, which causes a system crash.
IPv6 extension header	An attack sends the victim a packet with IPv6 extension headers.
IPv6 ext header abnormal	An attacker sends IPv6 packets with disordered or repeated IPv6 extension headers to the target.
IPv6 ext header exceed	An attacker sends IPv6 packets with IPv6 extension headers exceeding the upper limit to the target.

Scanning attacks

Scanning is a preintrusion activity used to prepare for intrusion into a network. The scanning allows the attacker to find a way into the target network and to disguise the attacker's identity.

Attackers use scanning tools to probe a network, find vulnerable hosts, and discover services that are running on the hosts. Attackers can use the information to launch attacks.

The device can detect and prevent the IP sweep and port scan attacks. If an attacker performs port scanning from multiple hosts to the target host, distributed port scan attacks occur.

Flood attacks

An attacker launches a flood attack by sending a large number of forged requests to the victim in a short period of time. The victim is too busy responding to these forged requests to provide services for legal users, and a DoS attack occurs.

The device can detect and prevent the following types of flood attacks.

SYN flood attack

A SYN flood attacker exploits the TCP three-way handshake characteristics and makes the victim unresponsive to legal users. An attacker sends a large number of SYN packets with forged source addresses to a server. This causes the server to open a large number of half-open connections and respond to the requests. However, the server will never receive the expected ACK packets. The server is unable to accept new incoming connection requests because all of its resources are bound to half-open connections.

ACK flood attack

An ACK packet is a TCP packet only with the ACK flag set. Upon receiving an ACK packet from a client, the server must search half-open connections for a match.

An ACK flood attacker sends a large number of ACK packets to the server. This causes the server to be busy searching for half-open connections, and the server is unable to process packets for normal services.

SYN-ACK flood attack

Upon receiving a SYN-ACK packet, the server must search for the matching SYN packet it has sent. A SYN-ACK flood attacker sends a large number of SYN-ACK packets to the server. This causes the server to be busy searching for SYN packets, and the server is unable to process packets for normal services.

FIN flood attack

FIN packets are used to shut down TCP connections.

A FIN flood attacker sends a large number of forged FIN packets to a server. The victim might shut down correct connections, or be unable to provide services because it is busy searching for matching connections.

RST flood attack

RST packets are used to abort TCP connections when TCP connection errors occur.

An RST flood attacker sends a large number of forged RST packets to a server. The victim might shut down correct connections, or be unable to provide services because it is busy searching for matching connections.

DNS flood attack

The DNS server processes and replies all DNS queries that it receives.

A DNS flood attacker sends a large number of forged DNS queries. This attack consumes the bandwidth and resources of the DNS server, which prevents the server from processing and replying legal DNS queries.

DNS response flood attack

The DNS client processes all DNS responses that it receives.

A DNS response flood attacker sends a large number of forged DNS responses. This attack consumes the bandwidth and resources of the DNS client, which prevents the client from processing legal DNS responses.

HTTP flood attack

Upon receiving an HTTP GET or POST request, the HTTP server performs complex operations, including character string searching, database traversal, data reassembly, and format switching. These operations consume a large amount of system resources.

An HTTP flood attacker sends a large number of HTTP GET or POST requests that exceed the processing capacity of the HTTP server, which causes the server to crash.

SIP flood attack

After receiving a SIP INVITE packet from a SIP client, the server must allocate resources to establish and trace the session with the SIP client.

A SIP flood attacker sends a large number of fake INVITE request packets at a rate exceeding the processing capacity of the SIP server, which causes the server to crash.

ICMP flood attack

An ICMP flood attacker sends ICMP request packets, such as ping packets, to a host at a fast rate. Because the target host is busy replying to these requests, it is unable to provide services.

ICMPv6 flood attack

An ICMPv6 flood attacker sends ICMPv6 request packets, such as ping packets, to a host at a fast rate. Because the target host is busy replying to these requests, it is unable to provide services.

UDP flood attack

A UDP flood attacker sends UDP packets to a host at a fast rate. These packets consume a large amount of the target host's bandwidth, so the host cannot provide other services.

Login DoS attack

In a login DoS attack, a malicious user can attempt to interfere with the normal operations of a device by flooding it with login requests. These requests consume the authentication resources, which makes the device unable to allow legal users to log in.

You can configure login attack prevention to prevent the login DoS attacks. This feature blocks user login attempts for a period of time after the user fails the maximum number of successive login attempts.

Login dictionary attack

The login dictionary attack is an automated process to attempt to log in by trying all possible passwords from a pre-arranged list of values (the dictionary). Multiple login attempts can occur in a short period of time.

You can configure the login delay feature to slow down the login dictionary attacks. This feature enables the device to delay accepting another login request after detecting a failed login attempt for a user.

HTTP slow attack

An attacker exploits the HTTP connection mechanism to establish a connection to an HTTP server and hold the connection for a long time in order to exhaust the server resources. The following types of HTTP slow attacks are commonly used:

- **Slow headers**—An attacker uses the HTTP GET or POST method to connect to the server. The HTTP header does not contain two CRLF sequences that mark the end of the header. In subsequent communication, the attacker sends packets to the server regularly with other HTTP header fields filled to keep the connection alive. The server is expecting the header end markers and maintains the connection for a long time.
- **Slow POST**—This type of attack occurs in one of the following conditions:
    - An attacker sends an HTTP POST request to submit data to the server and sets the **Content-Length** field to a greater value. In subsequent payload transisthmian, the attacker sends a small number of data each time to maintain the connection. The server keeps expecting the payload data from the attacker without releasing the connection.
    - An attacker sends an HTTP packet in chunked transfer encoding. If the HTTP packet is not ended with a zero-length chunk, the server is expecting the payload data from the attacker without releasing the connection.

To prevent HTTP slow attacks, configure HTTP slow attack detection and prevention. This feature detects such attacks and adds attacker addresses to the blacklist. Packets with source addresses on the blacklist are dropped.

Session creation attack

An attacker sends a large number of packets to create new sessions with the target to exhaust the target's resources and affect operation of its services.

To prevent session creation attacks, configure session creation rate limit to enable the device to limit the receiving rates of inbound packets for new sessions.

Blacklist feature

IP blacklist

The IP blacklist feature uses the source or destination IP addresses to filter packets.

- **Source IP blacklist**—Blocks packets if the source IP address of the packets matches a source IP blacklist entry.
- **Destination IP blacklist**—Blocks packets if the destination IP address of the packets matches a destination IP blacklist entry.

Compared with ACL-based packet filtering, IP blacklist filtering is simpler and provides effective screening at a faster speed.

User blacklist

The user blacklist feature is an attack prevention method that filters packets by source users in blacklist entries. Compared with IP blacklist filtering, user blacklist filtering performs access control on the user level and improves the filtering usability.

The user blacklist feature must be used together with the user identification feature. User identification provides the mappings between usernames and IP addresses for the user blacklist. For more information about user identification, see "Configuring user identification."

Address object group blacklist

The address object group blacklist feature is an attack prevention method that filters packets by address object group. The address object group blacklist feature must be used together with the address object group feature. An address object group is a set of IP address objects. For more information about address object groups, see "Configuring object groups." Compared with IP blacklist filtering, address object group blacklist filtering performs access control for subnets and improves the filtering usability.

Address object group whitelist

The address object group whitelist feature exempts packets from the whitelisted address object group from attack detection. Packets from the whitelisted address object group are directly forwarded whether they are attack packets or not. The address object group whitelist feature must be used together with the address object group feature. An address object group is a set of IP address objects. For more information about address object groups, see "Configuring object groups."

Client verification

TCP client verification

The TCP client verification feature protects TCP servers against the following flood attacks:

- SYN.
- ACK.
- SYN-ACK.
- FIN.
- RST.

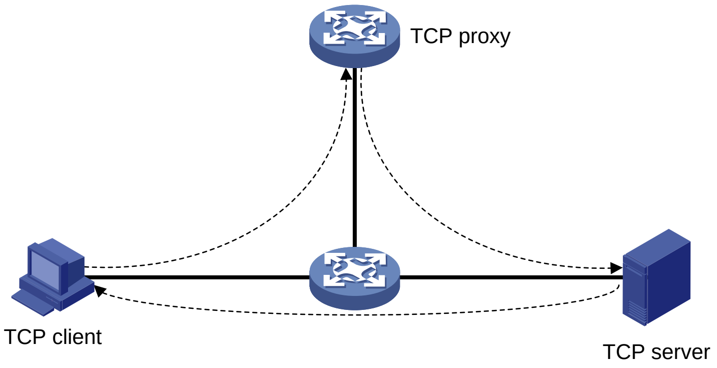
The TCP client verification feature enables a TCP proxy on the device.

TCP client verification can operate in the following modes:

- **Safe reset**—Enables unidirectional TCP proxy for packets only from TCP connection initiators. The unidirectional TCP proxy is sufficient for most scenarios because attacks are often seen from clients.

  As shown in Figure 1, if packets from TCP clients pass through the proxy device, but the packets from servers do not, only the safe reset mode can be used.

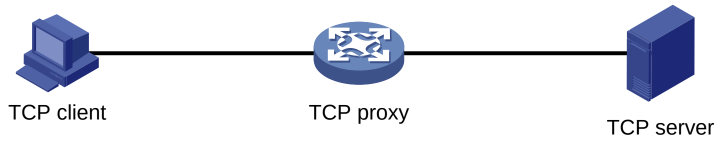
  Figure 1 Safe reset mode application



- **SYN cookie**—Enables bidirectional TCP proxy for TCP clients and servers.

As shown in Figure 2, if packets from clients and servers pass through the TCP proxy device, either safe reset or SYN cookie can be used.

Figure 2 Safe reset/SYN cookie mode application



TCP proxy in safe reset mode

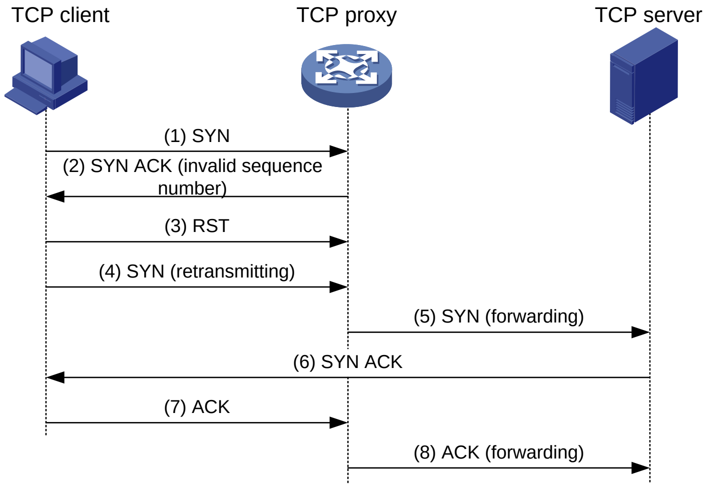
As shown in Figure 3, the safe reset mode functions as follows:

1. After receiving a SYN packet destined for a protected server, the TCP proxy sends back a SYN ACK packet with an invalid sequence number.
2. If the TCP proxy receives an RST packet from the client, the client is verified as legitimate.
3. The TCP proxy adds the client's IP address to the trusted IP list. The client initiates the connection again and the TCP proxy directly forwards the TCP packets to the server.

The safe reset mode requires that TCP clients comply with the TCP protocol suite. The TCP proxy will deny a legitimate client to access the server if the client does not comply with the TCP protocol suite.

With client verification, the TCP connection establishment takes more time than normal TCP connection establishment.

Figure 3 TCP proxy in safe reset mode



TCP proxy in SYN cookie mode

As shown in Figure 4, SYN cookie mode requires two TCP connections to be established as follows:

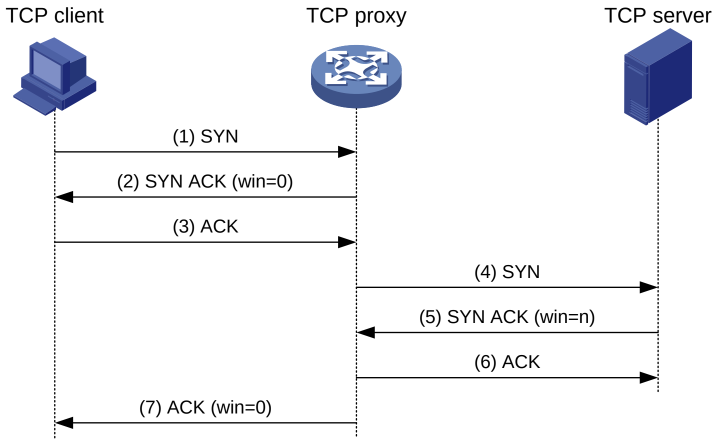
1. After receiving a SYN packet from a client to a protected server, the TCP proxy sends back a SYN ACK packet with the window size 0. If the client responds with an ACK packet, the client is verified as legitimate. The proxy device establishes a TCP connection with the client.
2. The TCP proxy device establishes a connection with the server through a new three-way handshake that has a different window size. This connection uses a different sequence number from the connection between the client and proxy device.

In SYN cookie mode, the TCP proxy is the server proxy that communicates with clients and the client proxy that communicates with server. Choose this mode when the following requirements are met:

- The TCP proxy device is deployed on the key path that passes through the ingress and egress of the protected server.

- All packets exchanged between clients and server pass through the TCP proxy device.

Figure 4 TCP proxy in SYN cookie mode



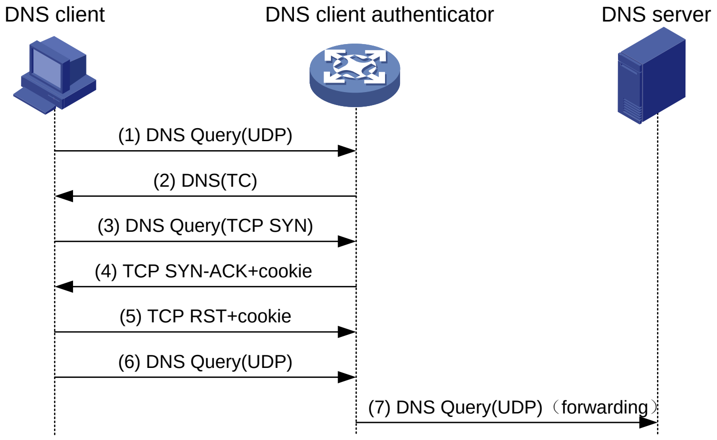
DNS client verification

The DNS client verification feature protects DNS servers against DNS flood attacks. It is configured on the device where packets from the DNS clients to the DNS servers pass through. The device with DNS client verification feature configured is called a DNS client authenticator.

As shown in Figure 5, the DNS client verification functions as follows:

1. Upon receiving a UDP DNS query destined for a protected server, the DNS client authenticator responds with a DNS truncate (TC) packet. The DNS truncate packet requires the client to initiate a query in a TCP packet.
2. When the authenticator receives a DNS query in a TCP SYN packet to port 53 from the client, the authenticator responds with a SYN-ACK packet that contains an incorrect sequence number.
3. When the authenticator receives a RST packet from the client, the authenticator verifies the client as legitimate.
4. The authenticator adds the client's IP address to the trusted IP list and forwards the trusted client's subsequent packets to the server.

Figure 5 DNS client verification process



The DNS client verification feature requires that clients use the standard TCP/IP protocol suite and DNS protocol. Legitimate clients that use non-standard protocols will be verified as illegitimate by the DNS client authenticator.

With client verification, the first DNS resolution takes more time than normal DNS resolution.

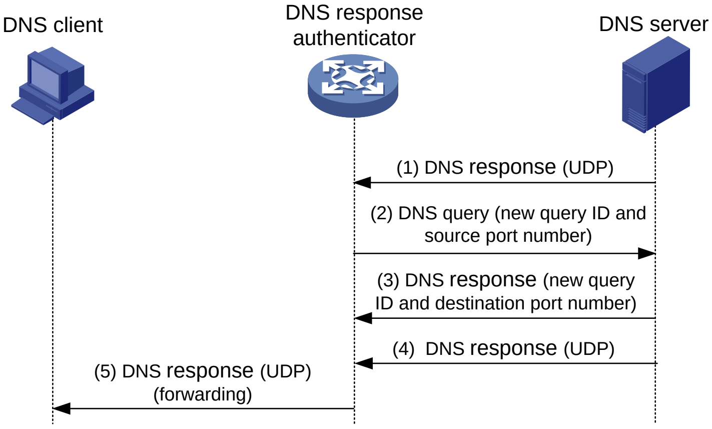
DNS response verification

The DNS response verification feature protects DNS clients against DNS response flood attacks. It is configured on the device where packets from the DNS servers to the DNS clients pass through. The device with DNS response verification feature configured is called a DNS response authenticator.

As shown in Figure 6, the DNS response verification functions as follows:

1. Upon receiving a UDP DNS response destined for a protected client, the DNS response authenticator sends back a DNS query packet with the locally generated query ID and port number.
2. After receiving the DNS query, a valid DNS server responds with a DNS response that contains a new query ID and destination port.
3. The DNS response authenticator verifies the query ID and destination port in the response. If the query ID and destination port are the same as the query ID and port number the authenticator has sent, the DNS server passes verification. The authenticator will forward subsequent packets from the server.

Figure 6 DNS response verification process


DNS client
DNS response authenticator
DNS server
(1) DNS response (UDP)
(2) DNS query (new query ID and source port number)
(3) DNS response (new query ID and destination port number)
(4) DNS response (UDP)
(5) DNS response (UDP) (forwarding)

HTTP client verification

The HTTP client verification feature protects HTTP servers against HTTP flood attacks. It is configured on the device where HTTP GET or POST request packets from the HTTP clients to the HTTP servers pass through. A device with HTTP client verification feature configured is called an HTTP client authenticator.

GET request-based verification

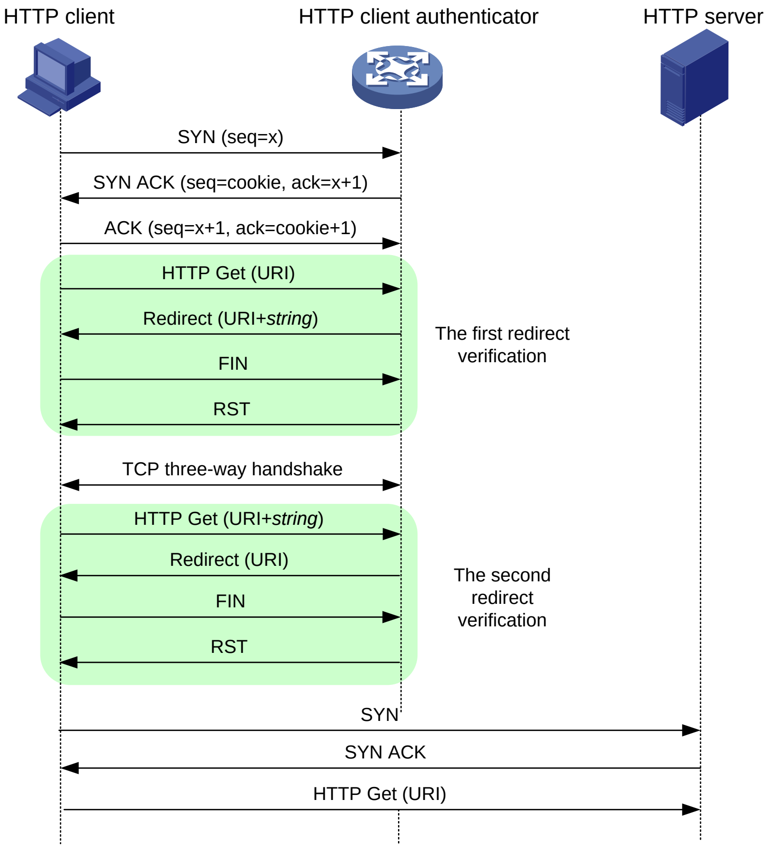
As shown in Figure 7, the HTTP client authenticator uses HTTP GET requests to verify the HTTP client as follows:

1. Upon receiving a SYN packet destined for a protected HTTP server, the HTTP client authenticator performs TCP client verification in SYN cookie mode. If the client passes the TCP client verification, a TCP connection is established between the client and the authenticator. For more information about TCP client verification, see "TCP client verification."
2. When the authenticator receives an HTTP GET packet from the client, it performs the first redirect verification. The authenticator records the client information and responds with an

HTTP Redirect packet. The HTTP Redirect packet contains a redirect URI and requires the client to terminate the TCP connection.

3. After receiving the HTTP Redirect packet, the client terminates the TCP connection and then establishes a new TCP connection with the authenticator.
4. When the authenticator receives the HTTP GET packet, it performs the second redirection verification. The authenticator verifies the following information:
   - The client has passed the first redirection verification.
   - The URI in the HTTP GET packet is the redirect URI.
5. If the client passes the second redirection verification, the authenticator adds its IP address to the trusted IP list, and responds a Redirect packet. The Redirect packet contains the URI that the client originally carried and requires the client to terminate the TCP connection.
6. The authenticator directly forwards the trusted client's subsequent packets to the server.

Figure 7 HTTP client verification process



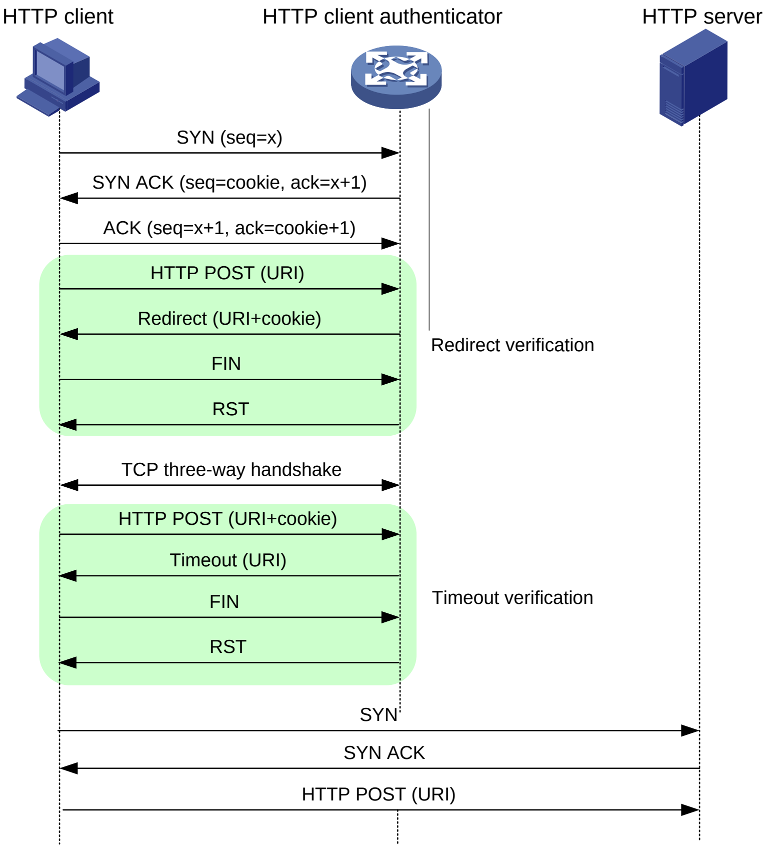
POST request-based verification

As shown in Figure 8, the HTTP client authenticator uses HTTP POST requests to verify the HTTP client as follows:

1. Upon receiving a SYN packet destined for a protected HTTP server, the HTTP client authenticator performs TCP client verification in SYN Cookie mode. If the client passes the TCP client verification, a TCP connection is established between the client and the authenticator. For more information about TCP client verification, see "TCP client verification."
2. When the authenticator receives an HTTP POST request from the client, it performs the redirect verification. The authenticator records the client information and responds with an HTTP Redirect packet. The HTTP Redirect packet contains a redirect URI and the Set-Cookie header, and requires the client to terminate the TCP connection.

3. After receiving the HTTP Redirect packet, the client terminates the TCP connection and then establishes a new TCP connection with the authenticator.
4. When the authenticator receives the HTTP POST request, it performs the timeout verification. The authenticator verifies the following information:
   - The client has passed the redirection verification.
   - The HTTP POST request contains a valid cookie.
5. If the client passes the timeout verification, the authenticator adds its IP address to the trusted IP list, and responds with an HTTP Timeout packet. The Timeout packet contains the URI that the client originally carried and requires the client to terminate the TCP connection.
6. The authenticator directly forwards the trusted client's subsequent packets to the server.

Figure 8 POST request-based verification process



SIP client verification

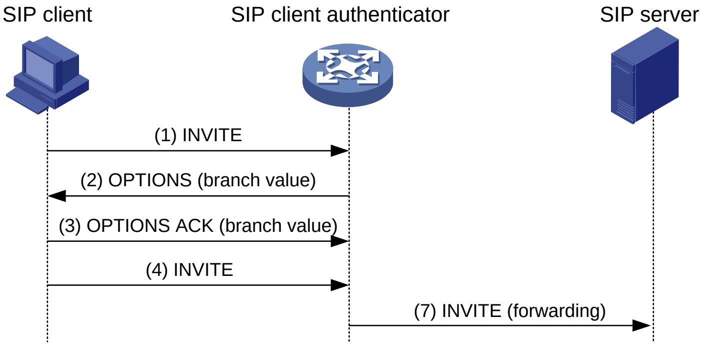
The SIP client verification feature protects SIP servers against SIP flood attacks. It is configured on the device where SIP INVITE request packets from the SIP clients to the SIP servers pass through. A device with the SIP client verification feature configured is called an SIP client authenticator.

As shown in Figure 9, the SIP client verification functions as follows:

1. Upon receiving a UDP INVITE packet destined for a protected server, the SIP client authenticator sends back an OPTIONS packet with a branch value.
2. After receiving the OPTIONS packet, the client sends an OPTIONS ACK to the SIP client authenticator.
3. When receiving the OPTIONS ACK, the SIP client authenticator verifies the branch value in the OPTIONS ACK.

- If the branch value in the OPTIONS ACK is the same as the branch value in the OPTIONS packet that the SIP client authenticator has sent, the client passes verification. The authenticator will forward subsequent packets from the client.
- If the branch value in the OPTIONS ACK is different from the branch value in the OPTIONS packet that the SIP client authenticator has sent, the client fails verification. The authenticator drops packets from the client.

Figure 9 SIP client verification process



Attack detection and prevention tasks at a glance

To configure attack detection and prevention, perform the following tasks:

1. Configuring and applying an attack defense policy
   a. Creating an attack defense policy
   b. Configuring an attack defense policy
      Choose the following tasks as needed:
      - Configuring a single-packet attack defense policy
      - Configuring a scanning attack defense policy
      - Configuring a flood attack defense policy
      - Configuring an HTTP slow attack defense policy
   c. (Optional.) Configuring attack detection exemption
   d. Applying an attack defense policy to a security zone
2. (Optional.) Configuring single-packet attack detection and prevention
3. (Optional.) Enabling log non-aggregation for single-packet attack events
4. (Optional.) Enabling the top attack statistics ranking feature
5. (Optional.) Configuring client verification
   Use this feature separately or jointly with a flood attack defense policy.
   - Configuring TCP client verification
   - Configuring DNS client verification
   - Configuring HTTP client verification
   - Configuring SIP client verification
6. (Optional.) Configuring the blacklist feature
   Use this feature separately or jointly with a scanning attack defense policy.
   - Configuring the IP blacklist feature
   - Configuring the user blacklist feature
   - Configuring the address object group blacklist

7. (Optional.) Configuring the address object group whitelist
8. (Optional.) Configuring the login attack prevention feature

   Typically, this feature is separately used.

   - Configuring login attack prevention
   - Enabling the login delay
9. (Optional.) Limiting the creation rate of new sessions

   Typically, this feature is separately used.
10. (Optional.) Configuring attack detection and prevention for a CPU core

Configuring and applying an attack defense policy

Creating an attack defense policy

About this task

An attack defense policy contains a set of attack detection and prevention configuration.

To configure attack defense configuration such as detection signatures and protection actions, you must first create an attack defense policy and enter its view.

Restrictions and guidelines

△ **CAUTION:**

The default thresholds for triggering attack prevention might not be appropriate for your network. Set appropriate thresholds according to the actual application scenarios. Small thresholds might affect the Internet or webpage access speed. Large thresholds might make your network vulnerable to attacks.

Procedure

1. Enter system view.

   `system-view`
2. Create an attack defense policy and enter its view.

   `attack-defense policy` *`policy-name`*

Configuring a single-packet attack defense policy

About this task

Apply the single-packet attack defense policy to the security zone that is connected to the external network.

Single-packet attack detection inspects incoming packets based on the packet signature. If an attack packet is detected, the device can take the following actions:

- Output logs (the default action).
- Drop attack packets.

You can also configure the device to not take any actions.

Restrictions and guidelines

The **`logging`** keyword enables the attack detection and prevention module to log single-packet attack events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output single-packet attack logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view single-packet attack logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Configure signature detection for specific single-packet attack types, and specify the actions against the attacks.

   - Configure signature detection for well-known single-packet attacks, and specify the actions against the attacks.

     signature detect { **fraggle** | **fragment** | **impossible** | **land** | **large-icmp** | **large-icmpv6** | **smurf** | **snork** | **tcp-all-flags** | **tcp-fin-only** | **tcp-invalid-flags** | **tcp-null-flag** | **tcp-syn-fin** | **tiny-fragment** | **traceroute** | **udp-bomb** | **winnuke** } [**action** { { **drop** | **logging** } * | **none** }]

     signature detect { **ip-option-abnormal** | **ping-of-death** | **teardrop** } **action** { **drop** | **logging** } *

   - Configure signature detection for ICMP packet attacks, and specify the actions against the attacks.

     signature detect icmp-type { *icmp-type-value* | **address-mask-reply** | **address-mask-request** | **destination-unreachable** | **echo-reply** | **echo-request** | **information-reply** | **information-request** | **parameter-problem** | **redirect** | **source-quench** | **time-exceeded** | **timestamp-reply** | **timestamp-request** } [**action** { { **drop** | **logging** } * | **none** }]

   - Configure signature detection for ICMPv6 packet attacks, and specify the actions against the attacks.

     signature detect icmpv6-type { *icmpv6-type-value* | **destination-unreachable** | **echo-reply** | **echo-request** | **group-query** | **group-reduction** | **group-report** | **packet-too-big** | **parameter-problem** | **time-exceeded** } [**action** { { **drop** | **logging** } * | **none** }]

   - Configure signature detection for IP option attacks, and specify the actions against the attacks.

     signature detect ip-option { *option-code* | **internet-timestamp** | **loose-source-routing** | **record-route** | **route-alert** | **security** | **stream-id** | **strict-source-routing** } [**action** { { **drop** | **logging** } * | **none** }]

   - Configure signature detection for IPv6 extension header attacks, and specify the actions against the attacks.

     signature detect ipv6-ext-header *ext-header-value* [**action** { { **drop** | **logging** } * | **none** }]

- Configure signature detection for abnormal IPv6 extension header attacks, and specify the actions against the attacks.

  signature detect ipv6-ext-header-abnormal [**action** { { **drop** | **logging** } * | **none** }]

- Configure signature detection for IPv6 extension header exceeded attacks, and specify the actions against the attacks.

  signature detect ipv6-ext-header-exceed [**limit** *limit-value*] [**action** { { **drop** | **logging** } * | **none** }]

By default, signature detection is not configured for single-packet attacks.

4. (Optional.) Set the maximum length of safe ICMP or ICMPv6 packets.

   signature { **large-icmp** | **large-icmpv6** } **max-length** *length*

   By default, the maximum length of safe ICMP or ICMPv6 packets is 4000 bytes.

5. (Optional.) Specify the actions against single-packet attacks of a specific level.

   signature level { **high** | **info** | **low** | **medium** } **action** { { **drop** | **logging** } * | **none** }

   The default action is **logging** for single-packet attacks of the informational and low levels.

   The default actions are **logging** and **drop** for single-packet attacks of the medium and high levels.

6. (Optional.) Enable signature detection for single-packet attacks of a specific level.

   signature level { **high** | **info** | **low** | **medium** } **detect**

   By default, signature detection is disabled for all levels of single-packet attacks.

Configuring a scanning attack defense policy

About this task

Apply a scanning attack defense policy to the security zone that is connected to the external network.

Scanning attack detection inspects the incoming packet rate of connections to the target system. If a source initiates connections at a rate equal to or exceeding the pre-defined threshold, the device can take the following actions:

- Output logs.
- Drop subsequent packets from the IP address of the attacker.
- Add the attacker's IP address to the IP blacklist.

If logging is specified for IP sweep and port scan attacks, the system outputs logs for only IP sweep attacks when both the IP sweep and port scan attack thresholds are reached.

Restrictions and guidelines

To blacklist the attackers, you must enable the blacklist feature globally or on the security zone where the defense policy is applied. For more information about the blacklist, see "Configuring the IP blacklist feature."

The **logging** keyword enables the attack detection and prevention module to log scanning attack events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output scanning attack logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view scanning attack logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Configure scanning attack detection.

   scan detect level { { **high** | **low** | **medium** } | **user-defined** { **port-scan-threshold** *threshold-value* | **ip-sweep-threshold** *threshold-value* } * [**period** *period-value*] } **action** { { **block-source** [**timeout** *minutes*] | **drop** } | **logging** } *

   By default, scanning attack detection is not configured.

Configuring a flood attack defense policy

About this task

Apply a flood attack defense policy to the security zone that is connected to the external network to protect internal servers.

Flood attack detection monitors the rate at which connections are initiated to the internal servers.

The device supports the following flood attack prevention types:

- **Source-based flood attack prevention**—Monitors the receiving rate of packets on a per-source IP basis. When the receiving rate of packets originating from an IP address keeps reaching or exceeding the threshold, the device enters prevention state and takes the specified defensive actions. Supported defensive actions include logging and dropping packets that originate from this IP address. When the rate drops below the silence threshold (three-fourths of the threshold), the device returns to the attack detection state.
- **Destination-based flood attack prevention**—Monitors the receiving rate of packets on a per-destination IP basis. When the receiving rate of packets destined for an IP address keeps reaching or exceeding the threshold, the device enters prevention state and takes the specified actions. Supported defensive actions include logging, dropping subsequent packets destined for this IP address, and client verification. When the rate drops below the silence threshold (three-fourths of the threshold), the device returns to the attack detection state.

An appropriate threshold can effectively prevent attacks. If the global threshold for triggering flood attack prevention is too low, false positives might occur, causing performance degradation or packet loss. If the global threshold is too high, false negatives might occur, making the network defenseless. Therefore, it is a good practice to enable the threshold learning feature for the device to automatically learn the global threshold. This feature allows the device to learn the global threshold based on the traffic flows in the network as follows:

1. Monitors the packet receiving rate in the network.
2. Calculates the global threshold based on the peak rate learned within the threshold learning duration.

You can choose to manually apply the learned threshold or configure the device to automatically apply the learned threshold.

The threshold learning feature includes the following modes:

- **One-time learning**—The device performs threshold learning only once.
- **Periodic learning**—The device performs threshold learning at intervals. The most recent learned threshold always takes effect.

Restrictions and guidelines for flood attack detection and prevention

If a device has multiple service cards, the global trigger threshold you set takes effect on each service card. The global trigger threshold of the device is the product of multiplying the value you set by the service card quantity.

You can configure flood attack detection and prevention for a specific IP address. Only destination-based flood attack prevention supports specifying IP addresses in the current software version. For non-specific IP addresses, the device uses the global attack prevention settings.

The **logging** keyword enables the attack detection and prevention module to log flood attack events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output flood attack logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view flood attack logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Configuring a SYN flood attack defense policy

1. Enter system view.
   system-view
2. Enter attack defense policy view.
   attack-defense policy *policy-name*
3. Enable global SYN flood attack detection.
   syn-flood detect non-specific
   By default, global SYN flood attack detection is disabled.
4. Set the global threshold for triggering source-based SYN flood attack prevention.
   syn-flood source-threshold *threshold-value*
   The default setting is 1000.
5. Set the global threshold for triggering destination-based SYN flood attack prevention.
   syn-flood threshold *threshold-value*
   The default setting is 10000.
6. Specify global actions against SYN flood attacks.
   syn-flood action { **client-verify** | **drop** | **logging** } *
   By default, no global action is specified for SYN flood attacks.
7. Configure IP address-specific SYN flood attack detection.
   syn-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]
   By default, IP address-specific SYN flood attack detection is not configured.

Configuring an ACK flood attack defense policy

1. Enter system view.
   system-view
2. Enter attack defense policy view.
   attack-defense policy *policy-name*

3. Enable global ACK flood attack detection.

   ack-flood detect non-specific

   By default, global ACK flood attack detection is disabled.

4. Set the global threshold for triggering source-based ACK flood attack prevention.

   ack-flood source-threshold *threshold-value*

   The default setting is 40000.

5. Set the global threshold for triggering destination-based ACK flood attack prevention.

   ack-flood threshold *threshold-value*

   The default setting is 40000.

6. Specify global actions against ACK flood attacks.

   ack-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for ACK flood attacks.

7. Configure IP address-specific ACK flood attack detection.

   ack-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific ACK flood attack detection is not configured.

Configuring a SYN-ACK flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global SYN-ACK flood attack detection.

   syn-ack-flood detect non-specific

   By default, global SYN-ACK flood attack detection is disabled.

4. Set the global threshold for triggering source-based SYN-ACK flood attack prevention.

   syn-ack-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based SYN-ACK flood attack prevention.

   syn-ack-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against SYN-ACK flood attacks.

   syn-ack-flood action { **client-verify** | **drop** | **logging** }*

   By default, no global action is specified for SYN-ACK flood attacks.

7. Configure IP address-specific SYN-ACK flood attack detection.

   syn-ack-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific SYN-ACK flood attack detection is not configured.

Configuring a FIN flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global FIN flood attack detection.

   fin-flood detect non-specific

   By default, global FIN flood attack detection is disabled.

4. Set the global threshold for triggering source-based FIN flood attack prevention.

   fin-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based FIN flood attack prevention.

   fin-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against FIN flood attacks.

   fin-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for FIN flood attacks.

7. Configure IP address-specific FIN flood attack detection.

   fin-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific FIN flood attack detection is not configured.

Configuring an RST flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global RST flood attack detection.

   rst-flood detect non-specific

   By default, global RST flood attack detection is disabled.

4. Set the global threshold for triggering source-based RST flood attack prevention.

   rst-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based RST flood attack prevention.

   rst-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against RST flood attacks.

   rst-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for RST flood attacks.

7. Configure IP address-specific RST flood attack detection.

   rst-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific RST flood attack detection is not configured.

Configuring an ICMP flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global ICMP flood attack detection.

   icmp-flood detect non-specific

   By default, global ICMP flood attack detection is disabled.

4. Set the global threshold for triggering source-based ICMP flood attack prevention.

   icmp-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based ICMP flood attack prevention.

   icmp-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against ICMP flood attacks.

   icmp-flood action { **drop** | **logging** } *

   By default, no global action is specified for ICMP flood attacks.

7. Configure IP address-specific ICMP flood attack detection.

   icmp-flood detect ip *ip-address* [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **drop** | **logging** } * | **none** }]

   By default, IP address-specific ICMP flood attack detection is not configured.

Configuring an ICMPv6 flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global ICMPv6 flood attack detection.

   icmpv6-flood detect non-specific

   By default, global ICMPv6 flood attack detection is disabled.

4. Set the global threshold for triggering source-based ICMPv6 flood attack prevention.

   icmpv6-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based ICMPv6 flood attack prevention.

   icmpv6-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against ICMPv6 flood attacks.

   icmpv6-flood action { **drop** | **logging** } *

   By default, no global action is specified for ICMPv6 flood attacks.

7. Configure IP address-specific ICMPv6 flood attack detection.

   icmpv6-flood detect ipv6 *ipv6-address* [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **drop** | **logging** } * | **none** }]

   By default, IP address-specific ICMPv6 flood attack detection is not configured.

Configuring a UDP flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global UDP flood attack detection.

udp-flood detect non-specific

By default, global UDP flood attack detection is disabled.

4. Set the global threshold for triggering source-based UDP flood attack prevention.

   udp-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based UDP flood attack prevention.

   udp-flood threshold *threshold-value*

   The default setting is 10000.

6. Specify global actions against UDP flood attacks.

   udp-flood action { **drop** | **logging** } *

   By default, no global action is specified for UDP flood attacks.

7. Configure IP address-specific UDP flood attack detection.

   udp-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **drop** | **logging** } * | **none** }]

   By default, IP address-specific UDP flood attack detection is not configured.

Configuring a DNS flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global DNS flood attack detection.

   dns-flood detect non-specific

   By default, global DNS flood attack detection is disabled.

4. Set the global threshold for triggering source-based DNS flood attack prevention.

   dns-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based DNS flood attack prevention.

   dns-flood threshold *threshold-value*

   The default setting is 10000.

6. (Optional.) Specify the global ports to be protected against DNS flood attacks.

   dns-flood port *port-list*

   By default, DNS flood attack prevention protects port 53.

7. Specify global actions against DNS flood attacks.

   dns-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for DNS flood attacks.

8. Configure IP address-specific DNS flood attack detection.

   dns-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-list*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific DNS flood attack detection is not configured.

Configuring a DNS response flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global DNS response flood attack detection.

   dns-reply-flood detect non-specific

   By default, global DNS response flood attack detection is disabled.

4. Set the global threshold for triggering source-based DNS response flood attack prevention.

   dns-reply-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based DNS response flood attack prevention.

   dns-reply-flood threshold *threshold-value*

   The default setting is 10000.

6. (Optional.) Specify the global ports to be protected against DNS response flood attacks.

   dns-reply-flood port *port-list*

   By default, DNS response flood attack prevention protects port 53.

7. Specify global actions against DNS response flood attacks.

   dns-reply-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for DNS response flood attacks.

8. Configure IP address-specific DNS response flood attack detection.

   dns-reply-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-list*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **logging** } * | **none** }]

   By default, IP address-specific DNS response flood attack detection is not configured.

Configuring an HTTP flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global HTTP flood attack detection.

   http-flood detect non-specific

   By default, global HTTP flood attack detection is disabled.

4. Set the global threshold for triggering source-based HTTP flood attack prevention.

   http-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based HTTP flood attack prevention.

   http-flood threshold *threshold-value*

   The default setting is 10000.

6. (Optional.) Specify the global ports to be protected against HTTP flood attacks.

   http-flood port *port-list*

   By default, HTTP flood attack prevention protects port 80.

7. Specify global actions against HTTP flood attacks.

   http-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for HTTP flood attacks.

8. Configure IP address-specific HTTP flood attack detection.

http-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-list*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

By default, IP address-specific HTTP flood attack detection is not configured.

Configuring a SIP flood attack defense policy

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global SIP flood attack detection.

   sip-flood detect non-specific

   By default, global SIP flood attack detection is disabled.

4. Set the global threshold for triggering source-based SIP flood attack prevention.

   sip-flood source-threshold *threshold-value*

   The default setting is 10000.

5. Set the global threshold for triggering destination-based SIP flood attack prevention.

   sip-flood threshold *threshold-value*

   The default setting is 10000.

6. (Optional.) Specify the global ports to be protected against SIP flood attacks.

   sip-flood port *port-list*

   By default, SIP flood attack prevention protects port 5060.

7. Specify global actions against SIP flood attacks.

   sip-flood action { **client-verify** | **drop** | **logging** } *

   By default, no global action is specified for SIP flood attacks.

8. Configure IP address-specific SIP flood attack detection.

   sip-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-list*] [**threshold** *threshold-value*] [**action** { { **client-verify** | **drop** | **logging** } * | **none** }]

   By default, IP address-specific SIP flood attack detection is not configured.

Configuring threshold learning for flood attack prevention

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable the threshold learning feature for flood attack prevention.

   threshold-learn enable

   By default, the threshold learning feature for flood attack prevention is disabled.

4. (Optional.) Set the threshold learning mode.

   - To set the one-time learning mode:
     threshold-learn mode once
   - To set the periodic learning mode:
     threshold-learn mode periodic

   By default, the one-time learning mode is used.

5. (Optional.) Set the threshold learning duration.

threshold-learn duration *duration*

By default, the threshold learning duration is 1440 minutes.

6. (Optional.) Set the threshold learning interval.

   threshold-learn interval *interval*

   By default, the threshold learning interval is 1440 minutes.

   Skip this step for the one-time learning mode.

7. (Optional.) Set the threshold learning tolerance value.

   threshold-learn tolerance-value *tolerance-value*

   By default, the threshold learning tolerance is 50, in percentage.

   Skip this step if auto application of the learned threshold is disabled.

8. (Optional.) Enable auto application of the learned threshold.

   threshold-learn auto-apply enable

   By default, auto application of the learned threshold is disabled.

9. Apply the most recent threshold that the device has learned.

   threshold-learn apply

   This command does not take effect when auto application of the learned threshold is enabled.

Configuring an HTTP slow attack defense policy

About this task

The device enters HTTP slow attack detection state when the number of HTTP concurrent connections reaches the detection triggering threshold. If the device receives an HTTP slow attack packet later, an HTTP slow attack occurs. When the number of HTTP slow attack packets exceeds the threshold within the detection period, the device takes defensive actions.

HTTP slow attack defensive actions include logging the attack events and blacklisting IP addresses of attackers.

Restrictions and guidelines

To use blacklisting as a defensive action, enable the blacklist feature.

As a best practice, specify port 80 as the global port to be protected against HTTP slow attacks. If you specify other ports by using the **http-slow-attack port** command, make sure these ports are used for HTTP communication. If the specified ports are not used for HTTP communication, the device resources will be wasted in inspecting non-HTTP slow attack packets.

Procedure

1. Enter system view.

   system-view

2. Enter attack defense policy view.

   attack-defense policy *policy-name*

3. Enable global HTTP slow attack detection.

   http-slow-attack detect non-specific

   By default, global HTTP slow attack detection is disabled.

4. Set the global thresholds for triggering HTTP slow attack prevention.

   http-slow-attack threshold [**alert-number** *alert-number* | **content-length** *content-length* | **payload-length** *payload-length* | **packet-number** *packet-number*]*

By default, thresholds for HTTP concurrent connections, the **Content-Length** field value, payload size, and abnormal packets are 5000, 10000, 50, and 10, respectively.

5. Set the global HTTP slow attack detection period.

   `http-slow-attack period` *`period`*

   By default, the global HTTP slow attack detection period is 60 seconds.

6. (Optional.) Specify the global ports to be protected against HTTP slow attacks.

   `http-slow-attack port` *`port-list`*

   By default, HTTP slow attack prevention protects port 80.

7. Specify global actions against HTTP slow attacks.

   `http-slow-attack action` { **`block-source`** [**`timeout`** *`minutes`*] | **`logging`** } *

   By default, no global action is specified for HTTP slow attacks.

8. Configure IP address-specific HTTP slow attack detection.

   `http-slow-attack detect` { **`ip`** *`ipv4-address`* | **`ipv6`** *`ipv6-address`* } [**`vpn-instance`** *`vpn-instance-name`*] [**`port`** *`port-list`*] [**`threshold`** { **`alert-number`** *`alert-number`* | **`content-length`** *`content-length`* | **`payload-length`** *`payload-length`* | **`packet-number`** *`packet-number`* }*] [**`period`** *`period`*] [**`action`** { **`block-source`** [**`timeout`** *`minutes`*] | **`logging`** }*]

   By default, IP address-specific HTTP slow attack detection is not configured.

Configuring attack detection exemption

About this task

The attack defense policy uses the ACL to identify exempted packets. The policy does not check the packets permitted by the ACL. You can configure the ACL to identify packets from trusted servers. The exemption feature reduces the false alarm rate and improves packet processing efficiency. For example, the attack defense policy identifies multicast packets with the same source addresses and different destination addresses as scanning attack packets (for example, OSPF or PIM packets). You can configure an ACL to exempt such packets from attack detection.

Restrictions and guidelines

If an ACL is used for attack detection exemption, only the following match criteria in the ACL permit rules take effect:

- Source IP address.
- Destination IP address.
- Source port.
- Destination port.
- Protocol.
- L3VPN instance.
- The **`fragment`** keyword for matching non-first fragments.

Procedure

1. Enter system view.

   `system-view`

2. Enter attack defense policy view.

   `attack-defense policy` *`policy-name`*

3. Configure attack detection exemption.

exempt acl [**ipv6**] { *acl-number* | **name** *acl-name* }

By default, attack detection exemption is not configured.

Applying an attack defense policy to a security zone

1. Enter system view.

   system-view

2. Enter security zone view.

   security-zone name *zone-name*

3. Apply an attack defense policy to the security zone.

   attack-defense apply policy *policy-name*

   By default, no attack defense policy is applied to the security zone.

Configuring single-packet attack detection and prevention globally

About this task

The global single-packet attack detection and prevention (also called the malformed packet attack detection and prevention) drops malformed packets of the following attacks on each interface:

- IP impossible packet attack.
- TCP packet attacks that use TCP packets with different flag settings (all flags set, only the FIN flag set, invalid flags, no flags set, and both SYN and FIN flags set).
- Land attack and WinNuke attack.
- UDP fraggle attack, UDP bomb attack, and UDP snork attack.

Restrictions and guidelines

The feature is more efficient than a single-packet attack defense policy in defend against malformed packet attacks. When this feature is enabled, you do not need to configure a single-packet attack defense policy to prevent attacks of the listed malformed packets.

Procedure

1. Enter system view.

   system-view

2. Enable malformed packet attack detection and prevention.

   attack-defense malformed-packet defend enable

   By default, malformed attack detection and prevention is enabled.

Enabling log non-aggregation for single-packet attack events

About this task

Log aggregation aggregates multiple logs generated during a period of time and sends one log. Logs that are aggregated must have the following attributes in common:

- Attacks are detected on the same security zone or are destined for the device.
- Attack type.

- Attack defense action.
- Source and destination IP addresses.
- VPN instance to which the victim IP address belongs.

Restrictions and guidelines

As a best practice, do not disable log aggregation. A large number of logs will consume the display resources of the console.

Procedure

1. Enter system view.
   `system-view`
2. Enable log non-aggregation for single-packet attack events.
   `attack-defense signature log non-aggregate`
   By default, log non-aggregation is disabled for single-packet attack events.

Enabling the top attack statistics ranking feature

About this task

This feature collects statistics about dropped attack packets based on attacker, victim, and attack type and ranks the top attack statistics by attacker and victim. To display the top attack statistics rankings, use the **`display attack-defense top-attack-statistics`** command.

Procedure

1. Enter system view.
   `system-view`
2. Enable the top attack statistics ranking feature.
   `attack-defense top-attack-statistics enable`
   By default, the top attack statistics ranking feature is disabled.

Configuring TCP client verification

About this task

Configure TCP client verification on the security zone that is connected to the external network. TCP client verification protects internal TCP servers against TCP flood attacks, including the following flood attacks:

- SYN.
- SYN-ACK.
- RST.
- FIN.
- ACK.

IP addresses protected by TCP client verification can be manually added or automatically learned:

- You can manually add protected IP addresses. The device performs client verification when it receives the first SYN packet destined for a protected IP address.
- The TCP client verification can automatically add victims' IP addresses to the protected IP list when collaborating with flood attack detection. Make sure **`client-verify`** is specified as the flood attack prevention action. For more information, see "Configuring a flood attack defense policy."

If a TCP client is verified legitimate in **safe reset** mode, the device adds the client's IP address to the trusted IP list. The device directly forwards TCP packets from trusted IP addresses.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Specify an IP address to be protected by the TCP client verification feature.

   client-verify tcp protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-number*]

3. Enter security zone view.

   security-zone name *zone-name*

4. Enable TCP client verification.

   - Set the safe reset mode.

     client-verify tcp enable mode safe-reset

   - Set the SYN cookie mode.

     client-verify tcp enable [**mode syn-cookie**]

   By default, TCP client verification is disabled.

Configuring DNS client verification

About this task

Configure DNS client verification on the security zone that is connected to the external network. The DNS client verification protects internal DNS servers against DNS flood attacks.

IP addresses protected by DNS client verification can be manually added or automatically learned:

- You can manually add protected IP addresses. The device performs client verification when it receives the first DNS query destined for a protected IP address.
- The DNS client verification can automatically add victims' IP addresses to the protected IP list when collaborating with DNS flood attack detection. Make sure **client-verify** is specified as the DNS flood attack prevention action. For more information, see "Configuring a DNS flood attack defense policy."

If a DNS client is verified legitimate, the device adds the client's IP address to the trusted IP list. The device directly forwards DNS packets from trusted IP addresses.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Specify an IP address to be protected by the DNS client verification feature.

   client-verify dns protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-number*]

3. Entersecurity zone view.

   security-zone name *zone-name*

4. Enable DNS client verification.

   client-verify dns enable

   By default, DNS client verification is disabled.

Configuring DNS response verification

About this task

Configure DNS response verification on the interface or security zone that is connected to the external network. The DNS response verification protects internal DNS clients against DNS response flood attacks.

IP addresses protected by DNS response verification can be manually added or automatically learned:

- You can manually add protected IP addresses. The device performs response verification when it receives the first DNS response destined for a protected IP address.
- The DNS response verification can automatically add victims' IP addresses to the protected IP list when collaborating with DNS response flood attack detection. Make sure **client-verify** is specified as the DNS response flood attack prevention action. For more information, see "Configuring a DNS response flood attack defense policy."

If a DNS server is verified legitimate, the device adds the client's IP address to the trusted IP list. The device directly forwards DNS responses from trusted IP addresses.

Restrictions and guidelines

The DNS response verification feature requires that servers use the standard TCP/IP protocol suite and DNS protocol. Legitimate servers that use non-standard protocols will be verified as illegitimate by the DNS response authenticator.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Specify an IP address to be protected by the DNS response verification feature.

   Client-verify dns-reply protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**port** *port-number*]

3. Enter interface/security zone view.

   interface *interface-type interface-number*

   security-zone name *zone-name*

4. Enable DNS response verification.

   client-verify dns-reply enable

   By default, DNS response verification is disabled.

Configuring HTTP client verification

About this task

Configure HTTP client verification on the security zone that is connected to the external network. The HTTP client verification protects internal HTTP servers against HTTP flood attacks.

IP addresses protected by HTTP client verification can be manually added or automatically learned:

- You can manually add protected IP addresses. The device performs client verification when it receives the first HTTP GET or POST packet destined for a protected IP address.
- The HTTP client verification can automatically add victims' IP addresses to the protected IP list when collaborating with HTTP flood attack detection. Make sure **client-verify** is specified as the HTTP flood attack prevention action. For more information, see "Configuring an HTTP flood attack defense policy."

If an HTTP client is verified legitimate, the device adds the client's IP address to the trusted IP list. The device directly forwards HTTP packets from trusted IP addresses.

Procedure

1. Enter system view.

   `system-view`

2. (Optional.) Specify an IP address to be protected by the HTTP client verification feature.

   `client-verify http protected` { **`ip`** *`destination-ip-address`* | **`ipv6`** *`destination-ipv6-address`* } [**`vpn-instance`** *`vpn-instance-name`*] [**`port`** *`port-number`*]

3. Enter security zone view.

   `security-zone name` *`zone-name`*

4. Enable HTTP client verification.

   `client-verify http enable`

   By default, HTTP client verification is disabled.

Configuring SIP client verification

About this task

Configure SIP client verification on the security zone that is connected to the external network. The SIP client verification protects internal SIP servers against SIP flood attacks.

IP addresses protected by SIP client verification can be manually added or automatically learned:

- You can manually add protected IP addresses. The device performs client verification when it receives the first INVITE packet destined for a protected IP address.
- The SIP client verification can automatically add victims' IP addresses to the protected IP list when collaborating with SIP flood attack detection. Make sure **client-verify** is specified as the SIP flood attack prevention action. For more information, see "Configuring a SIP flood attack defense policy."

If a SIP client is verified legitimate, the device adds the client's IP address to the trusted IP list. The device directly forwards SIP packets from trusted IP addresses.

Restrictions and guidelines

A legitimate SIP client might not pass the client verification if packets sent by the SIP client do not contain complete header information due to fragmentation.

Procedure

1. Enter system view.

   `system-view`

2. (Optional.) Specify an IP address to be protected by the SIP client verification feature.

   `client-verify sip protected` { **`ip`** *`destination-ip-address`* | **`ipv6`** *`destination-ipv6-address`* } [**`vpn-instance`** *`vpn-instance-name`*] [**`port`** *`port-number`*]

3. Enter security zone view.

   `security-zone name` *`zone-name`*

4. Enable SIP client verification.

   `client-verify sip enable`

   By default, SIP client verification is disabled.

Configuring the IP blacklist feature

About this task

The IP blacklist feature filters packets sourced from or destined for IP addresses in blacklist entries. If the global blacklist feature is enabled, the blacklist feature is enabled on all security zones.

You can manually add source or destination IP blacklist entries. When creating such an entry, you can set an aging time for it. Entries without the aging time do not age out unless you delete them manually.

The device can automatically add source IP blacklist entries when collaborating with scanning attack detection. Each dynamically learned source IP blacklist entry has an aging time, which is user configurable. Make sure the **block-source** keyword is specified as the scanning attack prevention action. For more information about the scanning attack detection and prevention, see "Configuring a scanning attack defense policy."

Procedure

1. Enter system view.

   system-view

2. (Optional.) Add an IP blacklist entry.
   - Add a source IPv4 blacklist entry.

     blacklist ip *source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*] [**timeout** *minutes*]
   - Add a source IPv6 blacklist entry.

     blacklist ipv6 *source-ipv6-address* [**vpn-instance** *vpn-instance-name*] [**timeout** *minutes*]
   - Add a destination IPv4 blacklist entry.

     blacklist destination-ip *destination-ip-address* [**vpn-instance** *vpn-instance-name*] [**timeout** *minutes*]
   - Add a destination IPv6 blacklist entry.

     blacklist destination-ipv6 *destination -ipv6-address* [**vpn-instance** *vpn-instance-name*] [**timeout** *minutes*]

3. (Optional.) Enable logging for the blacklist feature.

   blacklist logging enable

   By default, logging is disabled for the blacklist feature.

4. Enable the blacklist feature. Choose one option as needed:
   - Enable the global blacklist feature.

     blacklist global enable

     By default, the global blacklist feature is disabled.
   - Execute the following commands in sequence to enable the blacklist feature on a security zone:

     security-zone name *zone-name*

     blacklist enable

     By default, the blacklist feature is disabled on the /security zone.

Configuring the user blacklist feature

About this task

The user blacklist feature filters packets sourced from users in blacklist entries.

A user blacklist entry can only be manually added by using the **`blacklist user`** command. When creating such an entry, you can set an aging time for it. Entries without the aging time do not age out unless you delete them manually.

Restrictions and guidelines

The user blacklist feature must be used together with the user identification feature. For more information about user identification, see "Configuring user identification."

Procedure

1. Enter system view.

   `system-view`

2. Enable the global blacklist feature.

   `blacklist global enable`

   By default, the global blacklist feature is disabled.

3. Add a user blacklist entry.

   `blacklist user` *`user-name`* `[` **`domain`** *`domain-name`* `] [` **`timeout`** *`minutes`* `]`

4. (Optional.) Enable logging for the blacklist feature.

   `blacklist logging enable`

   By default, logging is disabled for the blacklist feature.

Configuring the address object group blacklist

About this task

This feature filters packets sourced from the subnets specified in the blacklisted address object group.

Restrictions and guidelines

An address object group can only be manually added to or deleted from the blacklist.

The address object group blacklist feature must be used together with the address object group feature. For more information about address object groups, see "Configuring object groups."

Procedure

1. Enter system view.

   `system-view`

2. Add an address object group to the blacklist.

   `blacklist object-group` *`object-group-name`*

   By default, no address object group is on the blacklist.

3. Enable the blacklist feature. Choose one option as needed:

   - Enable the global blacklist feature.

     `blacklist global enable`

     By default, the global blacklist feature is disabled.

   - Enter security zone view and enable the blacklist feature on the security zone.

     `security-zone name` *`zone-name`*

blacklist enable

By default, the blacklist feature is disabled on the security zone.

Configuring the address object group whitelist

About this task

This feature exempts packets sourced from the subnets specified in the whitelisted address object group from attack detection.

Restrictions and guidelines

An address object group can only be manually added to or deleted from the whitelist.

The address object group whitelist feature must be used together with the address object group feature. For more information about address object groups, see "Configuring object groups."

Procedure

1. Enter system view.

   system-view

2. Add an address object group to the whitelist.

   whitelist object-group *object-group-name*

   By default, no address object group is added to the whitelist.

3. Enable the whitelist feature. Choose one option as needed:

   - Enable the global whitelist feature.

     whitelist global enable

     By default, the global whitelist feature is disabled.

   - Enter security zone view and enable the whitelist feature on the security zone.

     security-zone name *zone-name*

     whitelist enable

     By default, the whitelist feature is disabled on the security zone.

Configuring login attack prevention

About this task

The login attack prevention feature detects a login DoS attack if a user fails the maximum number of successive login attempts. The feature triggers the blacklist feature to add the user's IP to the blacklist. Following login attempts from the user is blocked for the block period. For login attack prevention to take effect, you must enable the global blacklist feature.

This feature can effectively prevent login DoS attacks.

Restrictions and guidelines

The login attack prevention feature takes effect on users logging in through HTTP/HTTPS (including Web, NETCONF, AND RESTful), Telnet, terminal, SSH, and FTP. It does not support SNMP logins.

The login attack prevention feature takes effect on logins using local authentication or remote AAA server authentication.

To ensure that this feature can add a user's IP address to the blacklist after the user fails the maximum number of successive login attempts, complete the following configuration:

- Make sure the user's username exists on the device if the device performs local authentication for the user.

- Make sure the AAA server is reachable and well configured if the device uses remote AAA authentication for the user.

Procedure

1. Enter system view.

   `system-view`

2. Enable login attack prevention.

   `attack-defense login enable`

   By default, login attack prevention is disabled.

3. Set the maximum number of successive login failures.

   `attack-defense login max-attempt` *`max-attempt`*

   The default value is three.

4. Set the block period during which a login attempt is blocked.

   `attack-defense login block-timeout` *`minutes`*

   The default value is 60 minutes.

5. Enable the global blacklist feature.

   `blacklist global enable`

   By default, the global blacklist feature is disabled.

Enabling the login delay

About this task

The login delay feature delays the device from accepting a login request from a user after the user fails a login attempt. This feature can slow down login dictionary attacks.

The login delay feature is independent of the login attack prevention feature.

Procedure

1. Enter system view.

   `system-view`

2. Enable the login delay feature.

   `attack-defense login reauthentication-delay` *`seconds`*

   By default, the login delay feature is disabled. The device does not delay accepting a login request from a user who has failed a login attempt.

Limiting the creation rate of new sessions

About this task

This feature limits the receiving rate of inbound packets for new sessions to a specific value. The device supports rate limiting session creation based on the following criteria:

- Source IPv4 addresses.
- Source IPv6 addresses.
- Destination IPv4 addresses.
- Destination IPv6 addresses.

After you enable rate limit and specify a criterion on an interface, the device monitors the receiving rate of inbound packets for the matching sessions on this interface. When the receiving rate reaches or exceeds the rate limit, the device takes defense actions.

Restrictions and guidelines

You cannot enable session creation rate limit based on both source and destination IP addresses on the same interface.

Procedure

1. Enter system view

   system-view

2. Set defense actions upon threshold violations for monitored sessions.

   attack-defense ipcar { **destination** | **source** } { **ip** | **ipv6** } [**threshold** *threshold*] **action** { { **drop** | **logging** } * | **none** }

   By default, the packet receiving rate threshold is 5000 pps for each monitored session, and no defense actions are set.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable session creation rate limit.

   attack-defense ipcar { **destination** | **source** } { **ip** | **ipv6** } **session-rate-limit enable**

   By default, session creation rate limit is disabled.

Configuring attack detection and prevention for a CPU core

About this task

After the usage of a CPU core reaches the specified threshold and the shared queue of the driver is full, the system determines that an attack risk is present on the CPU core. Then, it processes the subsequent packets sent to the CPU core as follows:

- **Drop**—The CPU core uses all its available processing capability to process packets. The driver drops the packets beyond the maximum processing capability to decrease the CPU core usage. This action affects normal service processing.
- **Per-packet balance**—The CPU core uses all its available processing capability to process packets. Packets exceeding the maximum processing capability are sent to other CPU cores for load sharing on a per-packet basis. This action ensures normal service processing to some extent, but leads to risk of attacks on other CPU cores.
- **Isolate**—The driver isolates the flow that uses the most CPU time to lower the flow's processing priority. It sends the isolated packets to the CPU core for processing after the shared queue has no packets to process. This action ensures normal service processing to some extent, but it cannot significantly decrease the CPU usage because the packets in the public queue are still sent to the CPU core for processing.
- **No attack prevention action**—The driver takes no attack prevention action and still sends subsequent packets to the CPU core.

To set the CPU usage threshold per CPU core, execute the **context-capability inbound unicast total** command. For more information about this command, see context commands in *Virtual Technologies Command Reference*.

Hardware and feature compatibility

Models	Feature compatibility
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX5-HD6480	Yes

Models	Feature compatibility
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB680, NFNX3-HDB1080	No

Procedure

1. Enter system view
   system-view
2. Specify an attack prevention action for CPU core protection.
   attack-defense cpu-core action { **drop** | **isolate** | **per-packet-balance** }
   By default, the attack prevention action for CPU core protection is **drop**.

Display and maintenance commands for attack detection and prevention

Use the **display** commands in any view and the **reset** commands in user view.

To display and maintain attack detection and prevention:

Task	Command
Display flood attack detection and prevention statistics for an IPv4 address.	**display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **statistics ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**count**] [**security-zone** *zone-name*] [**slot** *slot-number*] [**count**]
Display flood attack detection and prevention statistics for an IPv6 address.	**display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **statistics ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**security-zone** *zone-name*] [**slot** *slot-number*] [**count**]
Display statistics about IPv4 HTTP slow attack detection and prevention.	**display attack-defense http-slow-attack statistics ip** [*ip-address* [**vpn-instance** *vpn-instance-name*]] [**security-zone** *zone-name*] [**slot** *slot-number*] [**count**]
Display statistics about IPv6 HTTP slow attack detection and prevention.	**display attack-defense http-slow-attack statistics ipv6** [*ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**security-zone** *zone-name*] [**slot** *slot-number*] [**count**]
Display statistics about malformed packets.	**display attack-defense malformed-packet statistics** [**slot** *slot-number*]
Display attack defense policy configuration.	**display attack-defense policy** [*policy-name*]
Display information about IPv4 addresses protected by flood attack	**display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** |

Task	Command
detection and prevention.	**fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display information about IPv6 addresses protected by flood attack detection and prevention.	**display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display information about IPv4 scanning attackers.	**display attack-defense scan attacker ip** [**security-zone** *zone-name* [**slot** *slot-number*]] [**count**]
Display information about IPv6 scanning attackers.	**display attack-defense scan attacker ipv6** [**security-zone** *zone-name* [**slot** *slot-number*]] [**count**]
Display attack detection and prevention statistics on a security zone.	**display attack-defense statistics security-zone** *zone-name* [**slot** *slot-number*]
Display top 10 attack statistics.	**display attack-defense top-attack-statistics** { **last-1-hour** | **last-24-hours** | **last-30-days** } [**by-attacker** | **by-type** | **by-victim**]
Display destination IPv4 blacklist entries.	**display blacklist destination-ip** [*destination-ip-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display destination IPv6 blacklist entries.	**display blacklist destination-ipv6** [*destination-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display source IPv4 blacklist entries.	**display blacklist ip** [*source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*]] [**slot** *slot-number*] [**count**]
Display source IPv6 blacklist entries.	**display blacklist ipv6** [*source-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display user blacklist entries.	**display blacklist user** [*user-name*] [**domain** *domain-name*] [**count**]
Display protected IPv4 list entries for client verification.	**display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*] [**slot** *slot-number*] [**count**]
Display protected IPv6 addresses for client verification.	**display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*]

Task	Command
	[**slot** *slot-number*] [**count**]
Display trusted IPv4 addresses for client verification.	**display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display trusted IPv6 addresses for client verification.	**display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
Display statistics about packets that match the address object groups on the whitelist.	**display whitelist object-group** [*object-group-name*] [**slot** *slot-number*]
Clear statistics about malformed packets.	**reset attack-defense malformed-packet statistics**
Clear flood attack detection and prevention statistics.	**reset attack-defense policy** *policy-name* **flood protected** { **ip** | **ipv6** } **statistics**
Clear attack detection and prevention statistics for a security zone.	**reset attack-defense statistics security-zone** *zone-name*
Clear top 10 attack statistics.	**reset attack-defense top-attack-statistics**
Delete dynamic destination IPv4 blacklist entries.	**reset blacklist destination-ip** { *destination-ip-address* [**vpn-instance** *vpn-instance-name*] | **all** }
Delete dynamic destination IPv6 blacklist entries.	**reset blacklist destination-ipv6**{ *destination-ipv6-address* [**vpn-instance** *vpn-instance-name*] | **all** }
Delete dynamic source IPv4 blacklist entries.	**reset blacklist ip** { *source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*] | **all** }
Delete dynamic source IPv6 blacklist entries.	**reset blacklist ipv6** { *source-ipv6-address* [**vpn-instance** *vpn-instance-name*] | **all** }
Clear blacklist statistics.	**reset blacklist statistics**
Clear protected IP statistics for client verification.	**reset client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected** { **ip** | **ipv6** } **statistics**
Clear the trusted IP list for client verification.	**reset client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted** { **ip** | **ipv6** }
Clear statistics about packets that match the address object groups on the whitelist.	**reset whitelist statistics**

Attack detection and prevention configuration examples

Example: Configuring security zone-based attack detection and prevention

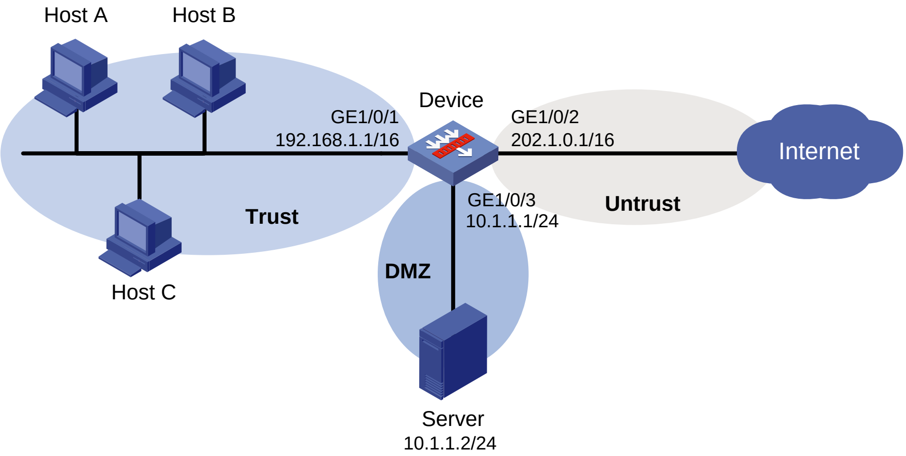
Network configuration

As shown in Figure 10, the device is the gateway for the internal network.

Configure an attack defense policy and apply the policy to security zone **Untrust** to meet the following requirements:

- Provide low-level scanning attack detection for the internal network. If a scanning attack is detected, log the attack and keep the attacker on the blacklist for 10 minutes.
- Protect internal hosts and servers against smurf attacks. If a smurf attack is detected, log the attack.
- Protect the internal server against SYN flood attacks. If the number of SYN packets sent to the server per second reaches or exceeds 5000, log the attack and drop subsequent packets.

Figure 10 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
```

```
[Device-security-zone-Untrust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
[Device-security-zone-DMZ] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow hosts in security zone **trust** to access the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   ```

   # Configure a rule named **untrust-dmz** to allow hosts on the Internet to access the server.

   ```
   [Device-security-policy-ip] rule name untrust-dmz
   [Device-security-policy-ip-2-untrust-dmz] source-zone untrust
   [Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
   [Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
   [Device-security-policy-ip-2-untrust-dmz] action pass
   [Device-security-policy-ip-2-untrust-dmz] quit
   [Device-security-policy-ip] quit
   ```

4. Configure an attack defense policy:

   # Create attack defense policy **a1**.

   ```
   [Device] attack-defense policy a1
   ```

   # Configure signature detection for smurf attacks, and specify **logging** as the prevention action.

   ```
   [Device-attack-defense-policy-a1] signature detect smurf action logging
   ```

   # Configure low-level scanning attack detection, specify **logging** and **block-source** as the prevention actions, and set the blacklist entry aging time to 10 minutes.

   ```
   [Device-attack-defense-policy-a1] scan detect level low action logging block-source
   timeout 10
   ```

   # Configure SYN flood attack detection for 10.1.1.2, set the attack prevention triggering threshold to 5000, and specify **logging** and **drop** as the prevention actions.

   ```
   [Device-attack-defense-policy-a1] syn-flood detect ip 10.1.1.2 threshold 5000 action
   logging drop
   [Device-attack-defense-policy-a1] quit
   ```

   # Apply attack defense policy **a1** to security zone **Untrust**.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] attack-defense apply policy a1
   [Device-security-zone-Untrust] quit
   ```

5. Enable the global blacklist feature.

   ```
   [Device] blacklist global enable
   ```

Verifying the configuration

# Verify that attack defense policy **a1** is successfully configured.

```
[Device] display attack-defense policy a1
          Attack-defense Policy Information
--------------------------------------------------------------------------
Policy name                        : a1
```

```
Applied list                        : Untrust
--------------------------------------------------------------------------
Exempt IPv4 ACL                     : Not configured
Exempt IPv6 ACL                     : Not configured
--------------------------------------------------------------------------
  Actions: CV-Client verify  BS-Block source  L-Logging  D-Drop  N-None

Signature attack defense configuration:
Signature name                      Defense      Level             Actions
Fragment                            Disabled     low               L
Impossible                          Disabled     medium            L,D
Teardrop                            Disabled     medium            L,D
Tiny fragment                       Disabled     low               L
IP option abnormal                  Disabled     medium            L,D
Smurf                               Enabled      medium            L
Traceroute                          Disabled     low               L
Ping of death                       Disabled     medium            L,D
Large ICMP                          Disabled     info              L
  Max length                        4000 bytes
Large ICMPv6                        Disabled     info              L
  Max length                        4000 bytes
TCP invalid flags                   Disabled     medium            L,D
TCP null flag                       Disabled     medium            L,D
TCP all flags                       Disabled     medium            L,D
TCP SYN-FIN flags                   Disabled     medium            L,D
TCP FIN only flag                   Disabled     medium            L,D
TCP Land                            Disabled     medium            L,D
Winnuke                             Disabled     medium            L,D
UDP Bomb                            Disabled     medium            L,D
UDP Snork                           Disabled     medium            L,D
UDP Fraggle                         Disabled     medium            L,D
IP option record route              Disabled     info              L
IP option internet timestamp        Disabled     info              L
IP option security                  Disabled     info              L
IP option loose source routing      Disabled     info              L
IP option stream ID                 Disabled     info              L
IP option strict source routing     Disabled     info              L
IP option route alert               Disabled     info              L
ICMP echo request                   Disabled     info              L
ICMP echo reply                     Disabled     info              L
ICMP source quench                  Disabled     info              L
ICMP destination unreachable        Disabled     info              L
ICMP redirect                       Disabled     info              L
ICMP time exceeded                  Disabled     info              L
ICMP parameter problem              Disabled     info              L
ICMP timestamp request              Disabled     info              L
ICMP timestamp reply                Disabled     info              L
ICMP information request            Disabled     info              L
ICMP information reply             Disabled     info              L
ICMP address mask request          Disabled     info              L
ICMP address mask reply            Disabled     info              L
ICMPv6 echo request                Disabled     info              L
ICMPv6 echo reply                  Disabled     info              L
ICMPv6 group membership query      Disabled     info              L
ICMPv6 group membership report     Disabled     info              L
ICMPv6 group membership reduction  Disabled     info              L
ICMPv6 destination unreachable     Disabled     info              L
ICMPv6 time exceeded               Disabled     info              L
ICMPv6 parameter problem           Disabled     info              L
ICMPv6 packet too big              Disabled     info              L
IPv6 extension header abnormal     Disabled     Info              L
IPv6 extension header exceeded     Disabled     Info              L
  Limit                            7


Scan attack defense configuration:
 Defense : Enabled
 Level   : low
 Actions : L,BS(10)

Flood attack defense configuration:
Flood type      Global thres(pps)  Global actions  Service ports   Non-specific
DNS flood        1000               -               53              Disabled
HTTP flood       1000               -               80              Disabled
SIP flood        1000               -               5060            Disabled
SYN flood        5000               L,D               -             Enabled
ACK flood        1000               -               -               Disabled
SYN-ACK flood    1000               -               -               Disabled
RST flood        1000               -               -               Disabled
FIN flood        1000               -               -               Disabled
UDP flood        1000               -               -               Disabled
ICMP flood       1000               -               -               Disabled
ICMPv6 flood     1000               -               -               Disabled


Flood attack defense for protected IP addresses:
 Address                 VPN instance Flood type    Thres(pps) Actions Ports
 10.1.1.2                --           SYN-FLOOD     5000       L,D     -
```

# Verify that the attack detection and prevention takes effect on security zone **Untrust**.

```
[Device] display attack-defense statistics security-zone untrust
Attack policy name: a1
Scan attack defense statistics:
 AttackType                          AttackTimes Dropped
 Port scan                           2           0
 IP sweep                            3           0
Flood attack defense statistics:
 AttackType                          AttackTimes Dropped
 SYN flood                           1           5000
Signature attack defense statistics:
 AttackType                          AttackTimes Dropped
 Smurf                               1           0
```

# Verify that the IPv4 blacklist collaborates with the scanning attack detection.

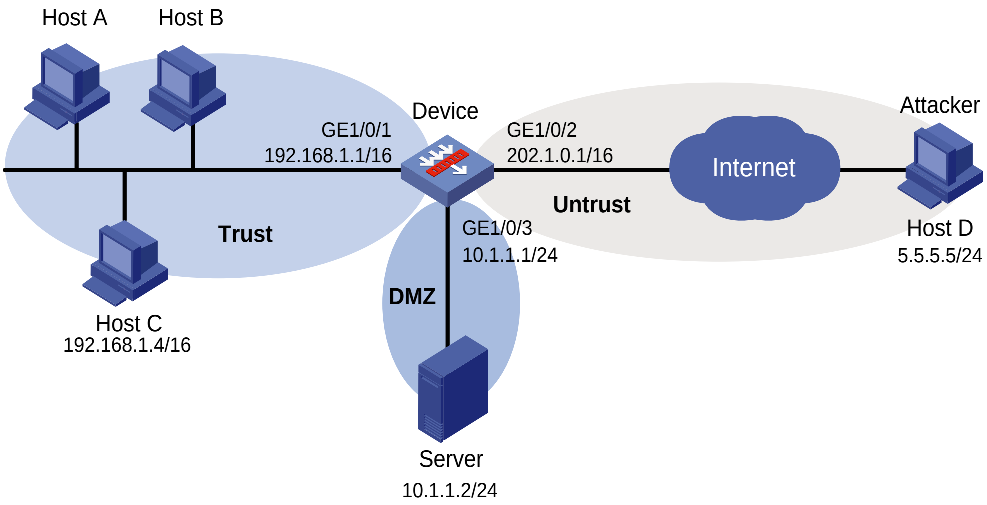
```
[Device] display blacklist ip
IP address      VPN instance   DS-Lite tunnel peer  Type    TTL(sec) Dropped
5.5.5.5         --             --                   Dynamic 600      353452
```

Example: Configuring the source IP blacklist

Network configuration

As shown in Figure 11, configure source IP blacklist entries on the device to block packets from the attacker Host D permanently and from Host C for 50 minutes.

Figure 11 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   ［Device］ security-zone name trust
   ［Device-security-zone-Trust］ import interface gigabitethernet 1/0/1
   ［Device-security-zone-Trust］ quit
   ［Device］ security-zone name untrust
   ［Device-security-zone-Untrust］ import interface gigabitethernet 1/0/2
   ［Device-security-zone-Untrust］ quit
   ［Device］ security-zone name dmz
   ```

```
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
[Device-security-zone-DMZ] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow hosts in security zone **trust** to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

   # Configure a rule named **untrust-dmz** to allow hosts on the Internet to access the server.

```
[Device-security-policy-ip] rule name untrust-dmz
[Device-security-policy-ip-2-untrust-dmz] source-zone untrust
[Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
[Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
[Device-security-policy-ip-2-untrust-dmz] action pass
[Device-security-policy-ip-2-untrust-dmz] quit
[Device-security-policy-ip] quit
```

4. Configure the source IP blacklist:

   # Add a source IPv4 blacklist entry for Host D.

```
[Device] blacklist ip 5.5.5.5
```

   # Add a source IPv4 blacklist entry for Host C and set the blacklist entry aging time to 50 minutes.

```
[Device] blacklist ip 192.168.1.4 timeout 50
```

   # Enable the global blacklist feature.

```
[Device] blacklist global enable
```

Verifying the configuration

# Verify that the source IPv4 blacklist entries are successfully added.

```
<Device> display blacklist ip
IP address      VPN instance   DS-Lite tunnel peer  Type    TTL(sec) Dropped
5.5.5.5         --             --                   Manual  Never    0
192.168.1.4     --             --                   Manual  2989     0
```

# Verify that the device drops packets from Host D. (Details not shown.)

# Execute the **undo blacklist ip 5.5.5.5** command and verify that the device forwards packets from Host D. (Details not shown.)

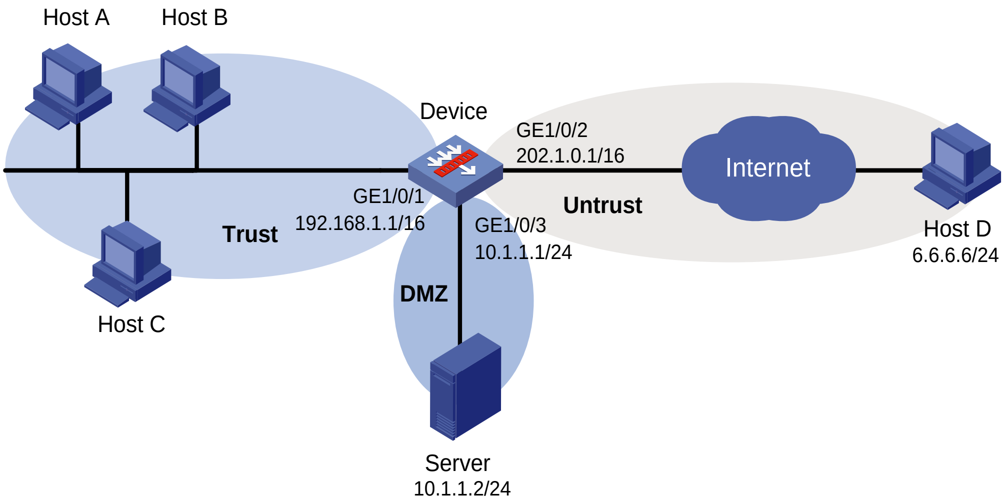
# Verify that the device drops packets from Host C for 50 minutes and forwards packets from Host C after 50 minutes. (Details not shown.)

Example: Configuring the destination IP blacklist

Network configuration

As shown in Figure 12, configure destination IP blacklist entries on the device to block packets destined for the server permanently and block packets destined for Host D for 50 minutes.

Figure 12 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
   [Device-security-zone-DMZ] quit
   ```

3. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow hosts in security zone **trust** to access the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   ```

   # Configure a rule named **untrust-dmz** to allow hosts on the Internet to access the server.

   ```
   [Device-security-policy-ip] rule name untrust-dmz
   [Device-security-policy-ip-2-untrust-dmz] source-zone untrust
   ```

```
[Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
[Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
[Device-security-policy-ip-2-untrust-dmz] action pass
[Device-security-policy-ip-2-untrust-dmz] quit
[Device-security-policy-ip] quit
```

4. Configure the destination IP blacklist:

   # Add a destination IPv4 blacklist entry for the server.

   ```
   [Device] blacklist destination-ip 10.1.1.2
   ```

   # Add a destination IPv4 blacklist entry for Host D and set the blacklist entry aging time to 50 minutes.

   ```
   [Device] blacklist destination-ip 6.6.6.6 timeout 50
   ```

   # Enable the global blacklist feature.

   ```
   [Device] blacklist global enable
   ```

Verifying the configuration

# Verify that the destination IPv4 blacklist entries are successfully added.

```
[Device] display blacklist destination-ip
IP address      VPN instance     Type    TTL(sec) Dropped
10.1.1.2        --               Manual  Never    0
6.6.6.6         --               Manual  2989     0
```

# Verify that the device drops packets destined for the server. (Details not shown.)

# Execute the **undo blacklist destination-ip 10.1.1.2** command and verify that the device forwards packets to the server. (Details not shown.)

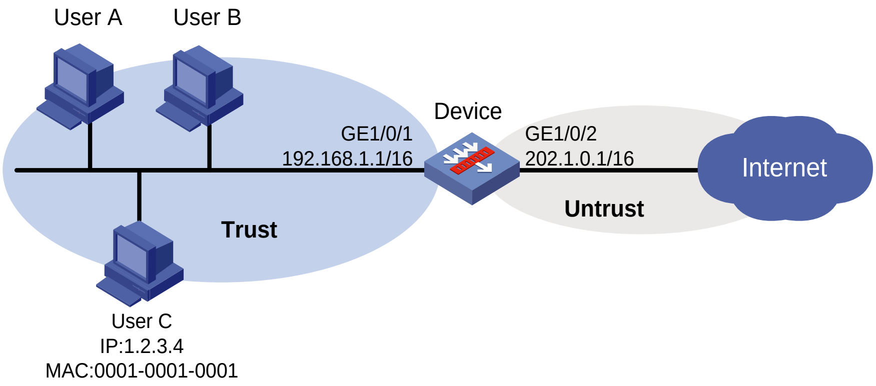
# Verify that the device drops packets destined for Host D for 50 minutes and forwards packets to Host B after 50 minutes. (Details not shown.)

Example: Configuring the user blacklist

Network configuration

As shown in Figure 13, configure the user blacklist feature on the device to block packets from User C for 50 minutes. The IP address of User C is 1.2.3.4 and the MAC address of User C is 0001-0001-0001.

Figure 13 Network diagram



Procedure

1. Assign IP addresses to interfaces:

Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure a security policy to allow hosts in security zone **trust** to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

4. Configure the user blacklist:

# Add a network access user named **userc**.

```
[Device] local-user userc class network
[Device-luser-network-userc] quit
```

# Configure a static identity user with the username **userc**, IP address 1.2.3.4, and MAC address 0001-0001-0001.

```
[Device] user-identity static-user userc bind ipv4 1.2.3.4 mac 0001-0001-0001
```

# Add a user blacklist entry for user **userc** and set the blacklist entry aging time to 50 minutes.

```
[Device] blacklist user userc timeout 50
```

# Enable user identification.

```
[Device] user-identity enable
```

# Enable the global blacklist feature.

```
[Device] blacklist global enable
```

Verifying the configuration

# Verify that the user blacklist entry is successfully added.

```
[Device] display blacklist user
User name   Domain name   Type    TTL(sec) Dropped
userc                     Manual  2987     0
```

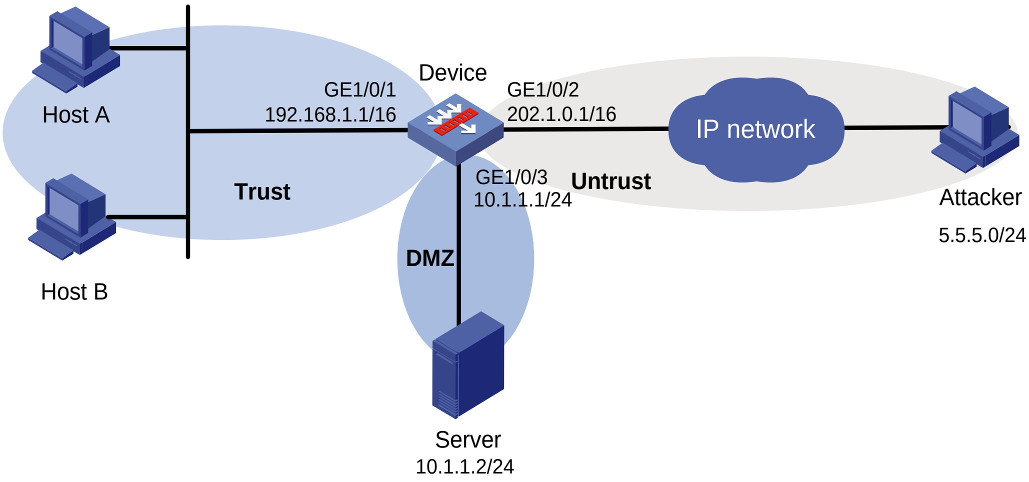
# Verify that the device drops packets from User C for 50 minutes and forwards packets from User C after 50 minutes. (Details not shown.)

Example: Configuring the address object group blacklist

Network configuration

As shown in Figure 14, configure the address object group blacklist feature on the device to block all packets from subnet 5.5.5.0/24 to prevent attacks from the subnet.

Figure 14 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
   [Device-security-zone-DMZ] quit
   ```

3. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow hosts in security zone **trust** to access the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
   [Device-security-policy-ip-1-trust-untrust] action pass
   ```

```
[Device-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **untrust-dmz** to allow hosts on the Internet to access the server.

```
[Device-security-policy-ip] rule name untrust-dmz
[Device-security-policy-ip-2-untrust-dmz] source-zone untrust
[Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
[Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
[Device-security-policy-ip-2-untrust-dmz] action pass
[Device-security-policy-ip-2-untrust-dmz] quit
[Device-security-policy-ip] quit
```

4. Configure the address object group blacklist:

# Create IPv4 address object group **obj1**. Configure an IPv4 address object with subnet 5.5.5.0/24.

```
[Device] object-group ip address obj1
[Device-obj-grp-ip-obj1] network subnet 5.5.5.0 24
[Device-obj-grp-ip-obj1] quit
```

# Add IPv4 address object group **obj1** to the blacklist.

```
[Device] blacklist object-group obj1
```

# Enable the global blacklist feature.

```
[Device] blacklist global enable
```

Verifying the configuration

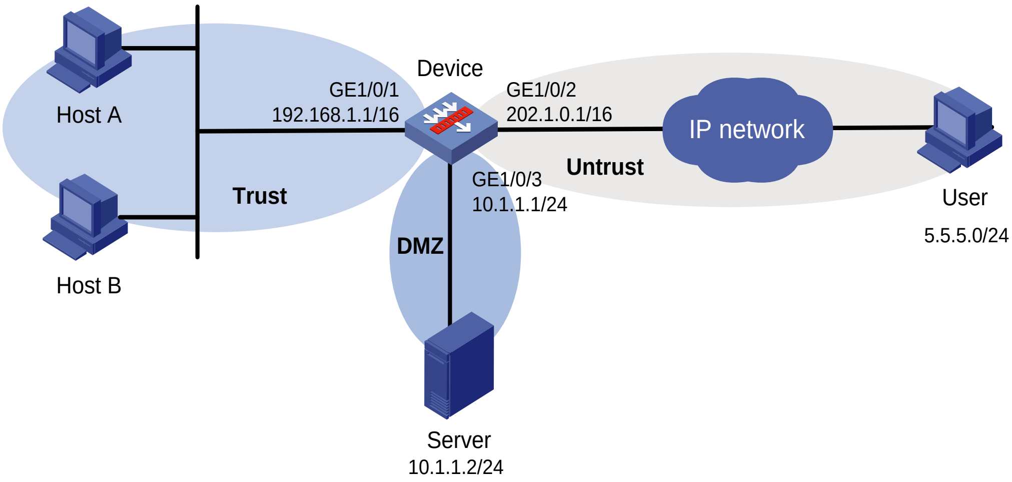
# Verify that the device drops all packets from subnet 5.5.5.0/24 unless you execute the **undo blacklist object-group** command on the device. (Details not shown.)

Example: Configuring the address object group whitelist

Network configuration

As shown in Figure 15, configure the address object group whitelist feature on the device to allow all packets from subnet 5.5.5.0/24 to pass through.

Figure 15 Network diagram



Procedure

1. Assign IP addresses to interfaces:

Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.0.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
[Device-security-zone-DMZ] quit
```

3. Configure a security policy:

# Configure a rule named **trust-untrust** to allow hosts in security zone **trust** to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.0.0 16
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **untrust-dmz** to allow hosts on the Internet to access the server.

```
[Device-security-policy-ip] rule name untrust-dmz
[Device-security-policy-ip-2-untrust-dmz] source-zone untrust
[Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
[Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
[Device-security-policy-ip-2-untrust-dmz] action pass
[Device-security-policy-ip-2-untrust-dmz] quit
[Device-security-policy-ip] quit
```

4. Configure the address object group whitelist:

# Create IPv4 address object group **obj1**. Configure an IPv4 address object with subnet 5.5.5.0/24.

```
[Device] object-group ip address obj1
[Device-obj-grp-ip-obj1] network subnet 5.5.5.0 24
[Device-obj-grp-ip-obj1] quit
```

# Add IPv4 address object group **obj1** to the whitelist.

```
[Device] whitelist object-group obj1
```

# Enable the global whitelist feature.

```
[Device] whitelist global enable
```

Verifying the configuration

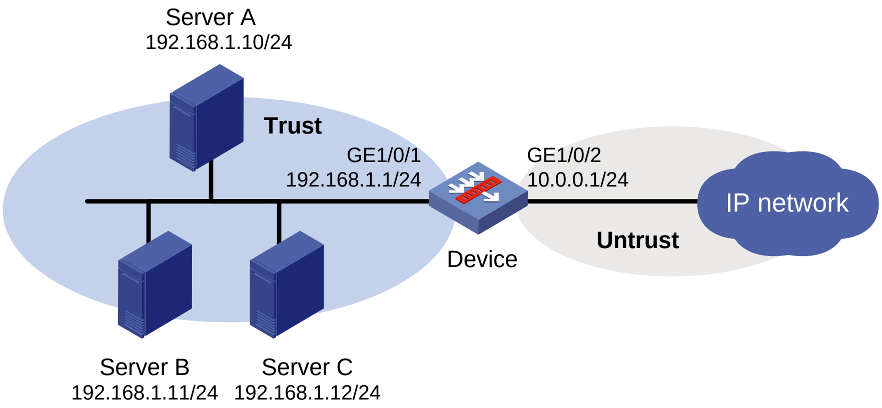
# Verify that the device allows all packets from subnet 5.5.5.0/24 to pass through unless you execute the **`undo whitelist object-group`** command on the device. (Details not shown.)

Example: Configuring security zone-based TCP client verification

Network configuration

As shown in Figure 16, configure TCP client verification in SYN cookie mode on the device to protect the internal servers against SYN flood attacks.

Figure 16 Network configuration



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy to allow hosts in security zone **trust** to access the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure TCP client verification:

# Create attack defense policy **a1**.

```
[Device] attack-defense policy a1
```

# Enable global SYN flood attack detection.

```
[Device-attack-defense-policy-a1] syn-flood detect non-specific
```

# Set the global threshold for triggering SYN flood attack prevention to 10000.

```
[Device-attack-defense-policy-a1] syn-flood threshold 10000
```

# Specify **logging** and **client-verify** as the global actions against SYN flood attacks.

```
[Device-attack-defense-policy-a1] syn-flood action logging client-verify
[Device-attack-defense-policy-a1] quit
```

# Apply attack defense policy **a1** to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] attack-defense apply policy a1
```

# Enable TCP client verification in SYN cookie mode on security zone **Untrust**.

```
[Device-security-zone-Untrust] client-verify tcp enable mode syn-cookie
[Device-security-zone-Untrust] quit
```

Verifying the configuration

# Launch a SYN flood attack. (Details not shown.)

# Verify that the victim's IP address is added to the protected IP list for TCP client verification.

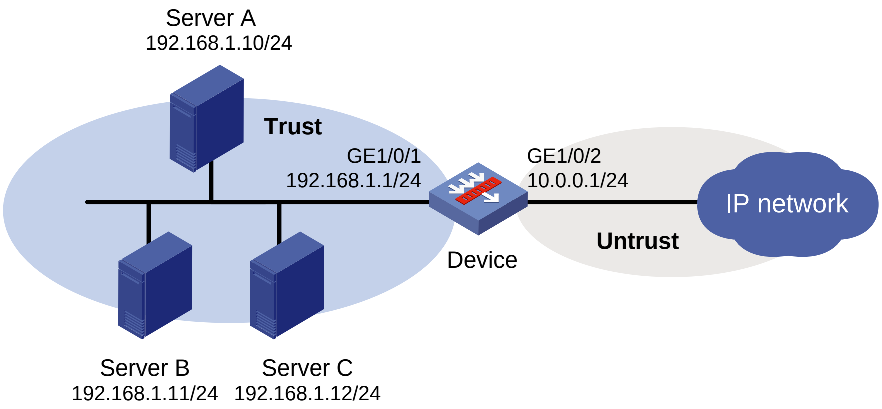
```
[Device] display client-verify tcp protected ip
IP address      VPN instance Port  Type     Requested  Trusted
192.168.1.10    --           any   Dynamic   20         12
```

Example: Configuring security zone-based DNS client verification

Network configuration

As shown in Figure 17, configure DNS client verification on the device to protect internal servers against DNS flood attacks.

Figure 17 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
```

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
［Device］ security-zone name trust
［Device-security-zone-Trust］ import interface gigabitethernet 1/0/1
［Device-security-zone-Trust］ quit
［Device］ security-zone name untrust
［Device-security-zone-Untrust］ import interface gigabitethernet 1/0/2
［Device-security-zone-Untrust］ quit
```

3. Configure a security policy to allow hosts in security zone **trust** to access the internal servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

4. Configure DNS client verification:

# Create attack defense policy **a1**.

```
[Device] attack-defense policy a1
```

# Enable global DNS flood attack detection.

```
[Device-attack-defense-policy-a1] dns-flood detect non-specific
```

# Set the global threshold for triggering DNS flood attack prevention to 10000.

```
[Device-attack-defense-policy-a1] dns-flood threshold 10000
```

# Specify **logging** and **client-verify** as the global actions against DNS flood attacks.

```
[Device-attack-defense-policy-a1] dns-flood action logging client-verify
[Device-attack-defense-policy-a1] quit
```

# Apply attack defense policy **a1** to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] attack-defense apply policy a1
```

# Enable DNS client verification on security zone **Untrust**.

```
[Device-security-zone-untrust] client-verify dns enable
[Device-security-zone-Untrust] quit
```

Verifying the configuration

# Launch a DNS flood attack. (Details not shown.)

# Verify that the victim's IP address is added to the protected IP list for DNS client verification**.**

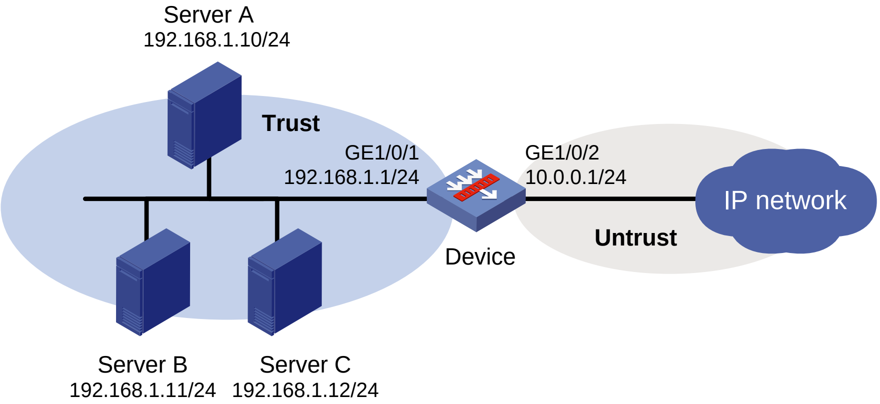
```
[Device] display client-verify dns protected ip
IP address      VPN instance Port  Type       Requested  Trusted
192.168.1.10    --           53    Dynamic    20         12
```

Example: Configuring security zone-based HTTP client verification

Network configuration

As shown in Figure 18, configure HTTP client verification on the device to protect internal servers against HTTP flood attacks.

Figure 18 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy to allow hosts in security zone **trust** to access the internal servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure HTTP client verification:

# Create attack defense policy **a1**.

```
[Device] attack-defense policy a1
```

# Enable global HTTP flood attack detection.

```
[Device-attack-defense-policy-a1] http-flood detect non-specific
```

# Set the global threshold for triggering HTTP flood attack prevention to 10000.

```
[Device-attack-defense-policy-a1] http-flood threshold 10000
```

# Specify **logging** and **client-verify** as the global actions against HTTP flood attacks.

```
[Device-attack-defense-policy-a1] http-flood action logging client-verify
[Device-attack-defense-policy-a1] quit
```

# Apply attack defense policy **a1** to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] attack-defense apply policy a1
```

# Enable HTTP client verification on security zone **Untrust**.

```
[Device-security-zone-Untrust] client-verify http enable
[Device-security-zone-Untrust] quit
```

Verifying the configuration

# Launch an HTTP flood attack. (Details not shown.)

# Verify that the victim's IP address is added to the protected IP list for HTTP client verification**.**

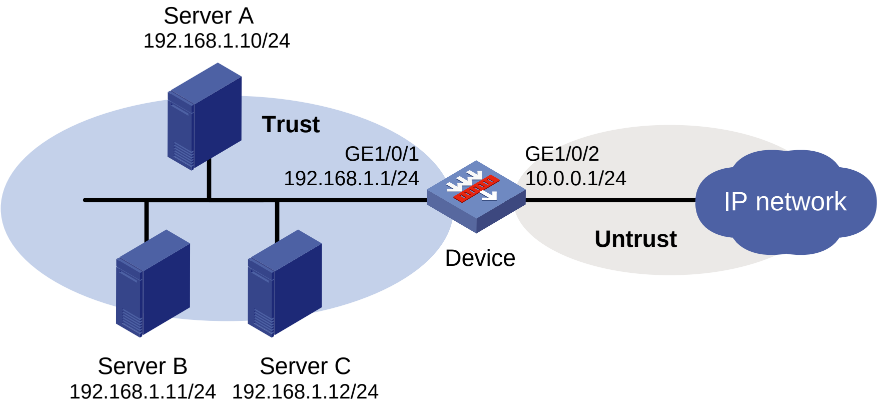
```
[Device] display client-verify http protected ip
IP address      VPN instance Port  Type       Requested  Trusted
192.168.1.10    --           8080  Dynamic    20         12
```

Example: Configuring security zone-based SIP client verification

Network configuration

As shown in Figure 19, configure SIP client verification on the device to protect internal servers against SIP flood attacks.

Figure 19 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
```

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure a security policy to allow hosts on the Internet to access the internal servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

4. Configure SIP client verification:

# Create attack defense policy **a1**.

```
[Device] attack-defense policy a1
```

# Enable global SIP flood attack detection.

```
[Device-attack-defense-policy-a1] sip-flood detect non-specific
```

# Set the global threshold to 10000 for triggering SIP flood attack prevention.

```
[Device-attack-defense-policy-a1] sip-flood threshold 10000
```

# Specify **logging** and **client-verify** as the global actions against SIP flood attacks.

```
[Device-attack-defense-policy-a1] sip-flood action logging client-verify
[Device-attack-defense-policy-a1] quit
```

# Apply attack defense policy **a1** to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] attack-defense apply policy a1
```

# Enable SIP client verification on security zone **Untrust**.

```
[Device-security-zone-Untrust] client-verify sip enable
[Device-security-zone-Untrust] quit
```

Verifying the configuration

# Launch a SIP flood attack. (Details not shown.)

# Verify that the victim's IP address is added to the protected IP list for SIP client verification**.**

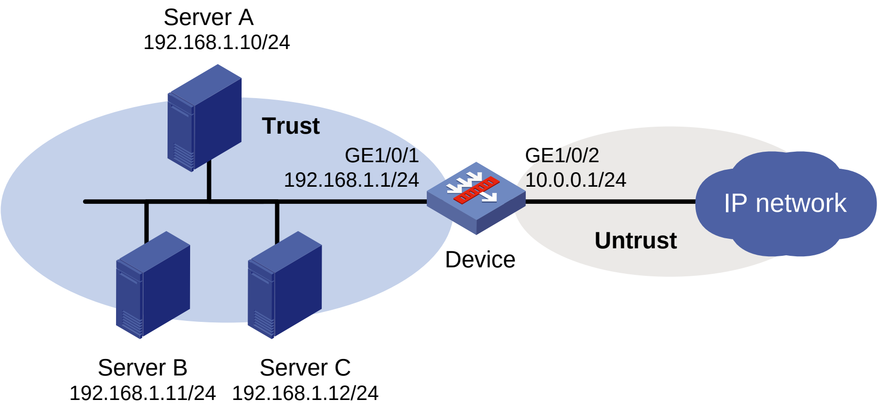
```
[Device] display client-verify sip protected ip
IP address      VPN instance Port  Type       Requested  Trusted
192.168.1.10    --           5060  Dynamic    20         12
```

Example: Configuring threshold learning for flood attack prevention

Network configuration

As shown in Figure 20, configure threshold learning for flood attack prevention on the device to protect internal servers against SYN flood attacks.

Figure 20 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy to allow hosts on the Internet to access the internal servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure threshold learning for flood attack prevention:

# Create attack defense policy **a1**.

```
[Device] attack-defense policy a1
```

# Set the threshold learning duration to 60 minutes.

```
[Device-attack-defense-policy-a1] threshold-learn duration 60
```

# Set the periodic learning mode.

```
[Device-attack-defense-policy-a1] threshold-learn mode periodic
```

# Set the threshold learning interval to 120 minutes.

```
[Device-attack-defense-policy-a1] threshold-learn interval 120
```

# Enable auto application of the learned threshold.

```
[Device-attack-defense-policy-a1] threshold-learn auto-apply enable
```

# Set the threshold learning tolerance value to 100.

```
[Device-attack-defense-policy-a1] threshold-learn tolerance-value 100
```

# Enable the threshold learning feature.

```
[Device-attack-defense-policy-a1] threshold-learn enable
```

# Enable global SYN flood attack detection.

```
[Device-attack-defense-policy-a1] syn-flood detect non-specific
```

# Specify **logging** and **drop** as the global actions against SYN flood attacks.

```
[Device-attack-defense-policy-a1] syn-flood action drop logging
[Device-attack-defense-policy-a1] quit
```

# Apply attack defense policy **a1** to security zone **Untrust**.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] attack-defense apply policy a1
```

Verifying the configuration

# Verify that the device has learned a threshold for triggering SYN flood attack prevention in 180 minutes (one learning cycle) or later.

```
[Device]display attack-defense policy a1
          Attack-defense Policy Information
--------------------------------------------------------------------------
Policy name                        : a1
Applied list                       : Untrust
--------------------------------------------------------------------------
Exempt IPv4 ACL                    : Not configured
Exempt IPv6 ACL                    : Not configured
--------------------------------------------------------------------------
  Actions: CV-Client verify  BS-Block source  L-Logging  D-Drop  N-None

Signature attack defense configuration:
Signature name                     Defense      Level             Actions
Fragment                           Disabled     low               L
Impossible                         Disabled     medium            L,D
Teardrop                           Disabled     medium            L,D
Tiny fragment                      Disabled     low               L
IP option abnormal                 Disabled     medium            L,D
Smurf                              Disabled     medium            L,D
Traceroute                         Disabled     low               L
Ping of death                      Disabled     medium            L,D
Large ICMP                         Disabled     info              L
```

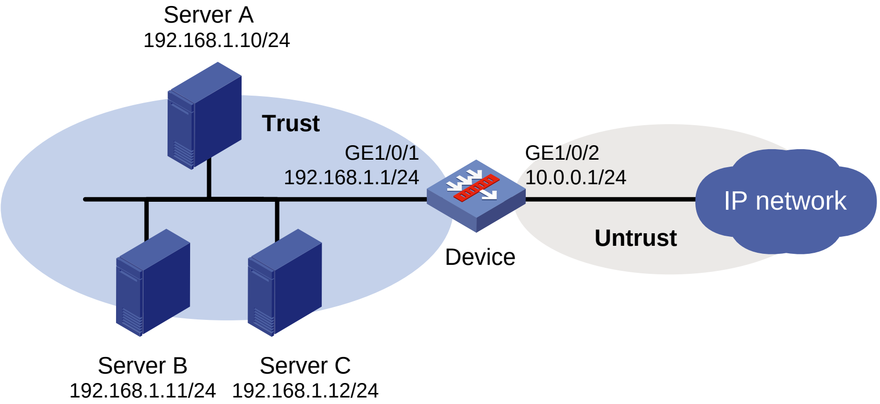
```
  Max length                        4000 bytes
Large ICMPv6                        Disabled     info              L
  Max length                        4000 bytes
TCP invalid flags                   Disabled     medium            L,D
TCP null flag                       Disabled     medium            L,D
TCP all flags                       Disabled     medium            L,D
TCP SYN-FIN flags                   Disabled     medium            L,D
TCP FIN only flag                   Disabled     medium            L,D
TCP Land                            Disabled     medium            L,D
Winnuke                             Disabled     medium            L,D
UDP Bomb                            Disabled     medium            L,D
UDP Snork                           Disabled     medium            L,D
UDP Fraggle                         Disabled     medium            L,D
IP option record route              Disabled     info              L
IP option internet timestamp        Disabled     info              L
IP option security                  Disabled     info              L
IP option loose source routing      Disabled     info              L
IP option stream ID                 Disabled     info              L
IP option strict source routing     Disabled     info              L
IP option route alert               Disabled     info              L
ICMP echo request                   Disabled     info              L
ICMP echo reply                     Disabled     info              L
ICMP source quench                  Disabled     info              L
ICMP destination unreachable        Disabled     info              L
ICMP redirect                       Disabled     info              L
ICMP time exceeded                  Disabled     info              L
ICMP parameter problem              Disabled     info              L
ICMP timestamp request              Disabled     info              L
ICMP timestamp reply                Disabled     info              L
ICMP information request            Disabled     info              L
ICMP information reply              Disabled     info              L
ICMP address mask request           Disabled     info              L
ICMP address mask reply             Disabled     info              L
ICMPv6 echo request                 Disabled     info              L
ICMPv6 echo reply                   Disabled     info              L
ICMPv6 group membership query       Disabled     info              L
ICMPv6 group membership report      Disabled     info              L
ICMPv6 group membership reduction   Disabled     info              L
ICMPv6 destination unreachable      Disabled     info              L
ICMPv6 time exceeded                Disabled     info              L
ICMPv6 parameter problem            Disabled     info              L
ICMPv6 packet too big               Disabled     info              L
IPv6 extension header abnormal      Disabled     Info              L
IPv6 extension header exceeded      Disabled     Info              L
  Limit                             7


Scan attack defense configuration:
 Defense : Disabled
 Level   : -
 Actions : -

Flood attack defense configuration:
Flood type      Global thres(pps)  Global actions  Service ports   Non-specific
DNS flood        1000               -               53              Disabled
HTTP flood       1000               -               80              Disabled
SIP flood        1000               -               5060            Disabled
SYN flood        1000               L,D               -             Enabled
ACK flood        1000               -               -               Disabled
SYN-ACK flood    1000               -               -               Disabled
RST flood        1000               -               -               Disabled
FIN flood        1000               -               -               Disabled
UDP flood        1000               -               -               Disabled
ICMP flood       1000               -               -               Disabled
ICMPv6 flood     1000               -               -               Disabled

Flood attack defense for protected IP addresses:
 Address                  VPN instance Flood type    Thres(pps) Actions Ports
```

Example: Limiting the creation rate of new sessions

Network configuration

As shown in Figure 21, limit the creation rate of new sessions on the device to protect the internal servers against external DDoS attacks.

Figure 21 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure a security policy to allow hosts on the Internet to access internal servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.10
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.11
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.12
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

4. Limit the creation rate of new sessions:

# Limit sessions on a per-source IPv4 address, set the packet receiving rate threshold to 10 pps and set the **drop** action.

```
[Device] attack-defense ipcar source ip threshold 10 action logging drop
```

# Enable session creation rate limit based on source IPv4 addresses on GigabitEthernet 2/0/1.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] attack-defense ipcar source ip session-rate-limit
enable
```

Contents

Configuring server connection detection

About server connection detection

Server connection detection (SCD) provides protections for internal servers. It enables the device to identify legal and illegal connections initiated by the protected servers. This helps you monitor internal servers and prevent them from becoming part of a botnet and launching attacks or performing internal network penetration.

SCD tasks at a glance

To configure SCD, perform the following tasks:

1. Configuring server connection learning
2. Configuring an SCD policy
3. Configuring SCD rules in an SCD policy

Configuring server connection learning

About this task

Server connection learning learns connections initiated by given servers. The learning results provide the basis for you to create SCD policies to monitor and log illegal connections initiated by the servers.

Restrictions and guidelines

You cannot edit any settings in server connection learning configuration view if server connection learning is in progress.

Procedure

1. Enter system view.
   system-view
2. Enter server connection learning configuration view.
   scd learning
3. Specify an IP address object group for server connection learning.
   source-ip *object-group-name*
   By default, no IP address object groups are specified for server connection learning.
4. Enable server connection learning for a learning period.
   auto-learn enable period { **one-day** | **one-hour** | **seven-day** | **twelve-hour** }
   By default, server connection learning is disabled.

Configuring an SCD policy

About this task

An SCD policy monitors the connections initiated by the specified protected server. You can configure the following settings in an SCD policy:

- Protected server IP address.
- SCD rules to identify legal connections initiated by the server.
- Logging for illegal connections initiated by the server.
- SCD policy enabling status.

Restrictions and guidelines

An SCD policy monitors only the connections initiated by the server specified by the **protected-server** command.

The protected server IP address must be unique for each SCD policy.

Procedure

1. Enter system view.
   system-view
2. Create an SCD policy and enter its view.
   scd policy name *policy-name*
3. Specify the IP address of the protected server.
   protected-server *ip-address*

   By default, no protected server IP address is specified.
4. (Optional.) Enable logging for illegal connections initiated by the protected server.
   logging enable

   By default, the device does not log illegal connections initiated by the protected server.
5. Enable the SCD policy.
   policy enable

   By default, an SCD policy is disabled.

Configuring SCD rules in an SCD policy

About this task

You can configure multiple SCD rules in an SCD policy. Each SCD rule contains the following criteria to identify legal connections initiated by the protected server:

- A destination IP address criterion, which specifies the destination IP address for server-initiated connections.
- One or more protocol criteria. Each protocol criterion specifies a protocol and optionally a set of destination port numbers.

A connection initiated by the protected server matches the SCD rule if the connection matches both the destination IP address criterion and a protocol criterion. Connections initiated by the server that do not match any SCD rules are considered illegal connections.

Restrictions and guidelines

If you do not configure any rules in an SCD policy, all connections initiated by the protected server of the policy are illegal connections.

An SCD rule must contain one destination IP address criterion and a minimum of one protocol criteria.

In one SCD policy, each SCD rule must use a unique destination IP address.

Procedure

1. Enter system view.

```
system-view
```

2. Enter the view of an SCD policy.

```
scd policy name policy-name
```

3. Create an SCD rule and enter its view.

```
rule rule-id
```

4. Configure the destination IP address criterion.

```
permit-dest-ip ip-address
```

By default, the destination IP address criterion is not configured.

5. Configure a protocol criterion.

```
protocol { icmp | tcp port port-list | udp port port-list }
```

By default, no protocol criterion is configured in an SCD rule.

Display and maintenance commands for server connection detection

Execute **display** commands in any view.

Execute **reset** commands in user view.

Task	Command
Display the server connection learning information.	**display scd auto-learn config**
Display the server connection learning results.	**display scd learning record** [**protected-server** *ip-address*] [**destination-ip** *ip-address*]
Display the SCD policy information.	**display scd policy** [**name** *policy-name*]
Clear the server connection learning results.	**reset scd learning record**

SCD configuration examples

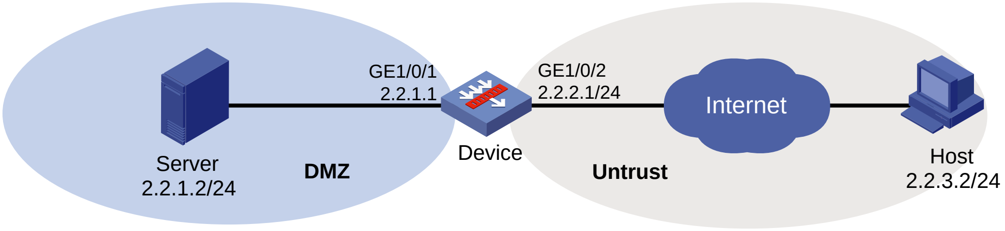
Example: Configuring SCD

Network configuration

As shown in Figure 1, configure SCD on the device to perform the following tasks:

- Monitor connections initiated by servers in subnet 2.2.1.0/24 for one day.
- Logs all connections initiated by the server except for TCP connections destined for TCP ports 80 and 443 on host 2.2.3.2/24.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 2.2.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 2.2.3.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   [Device-security-zone-DMZ] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **dmz-untrust** to permit the packets from the internal server to the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name dmz-untrust
   [Device-security-policy-ip-1-dmz-untrust] source-zone dmz
   [Device-security-policy-ip-1-dmz-untrust] destination-zone untrust
   [Device-security-policy-ip-1-dmz-untrust] source-ip-host 2.2.1.2
   [Device-security-policy-ip-1-dmz-untrust] action pass
   [Device-security-policy-ip-1-dmz-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Create an IP address object group named **abc** and specify IPv4 address 2.2.1.0 and mask length 24 for the object group.

   ```
   [Device] object-group ip address abc
   [Device-obj-grp-ip-abc] network subnet 2.2.1.0 24
   [Device-obj-grp-ip-abc] quit
   ```

6. Specify IP address object group **abc** for server connection learning and enable server connection learning for one day.

   ```
   [Device] scd learning
   [Device-scd-learning] source-ip abc
   [Device-scd-learning] auto-learn enable period one-day
   ```

```
[Device-scd-learning] quit
```

7. Create an SCD policy named **policy1**.

```
[Device] scd policy name policy1
[Device-scd-policy-policy1] protected-server 2.2.1.2
[Device-scd-policy-policy1] logging enable
[Device-scd-policy-policy1] rule 1
[Device-scd-policy-policy1-1] permit-dest-ip 2.2.3.2
[Device-scd-policy-policy1-1] protocol tcp port 80 443
[Device-scd-policy-policy1-1] quit
[Device-scd-policy-policy1] policy enable
[Device-scd-policy-policy1] quit
```

Verifying the configuration

# After the server connection learning is complete, display the server connection learning results.

```
[Device] display scd learning record
Id      Protected server    Destination IPv4 address   Protocol    Port
1       2.2.1.2             2.2.3.2                    TCP         80
2       2.2.1.2             2.2.3.2                    TCP         443
3       2.2.1.2             2.2.3.2                    UDP         4433
4       2.2.1.2             2.2.3.2                    UDP         567
Total entries: 4
```

# Display information about SCD policy **policy1**.

```
<Sysname> display scd policy name policy1
SCD policy name: policy1
 Protected server IPv4: 2.2.1.2
 Logging: Enabled
 Policy status: Enabled
 Rule ID: 1
  Permitted dest IPv4: 2.2.3.2
  Protocol: TCP port 80,443
```

Contents

Configuring ARP attack protection

About ARP attack protection

The device can provide multiple features to detect and prevent ARP attacks and viruses in the LAN. An attacker can exploit ARP vulnerabilities to attack network devices in the following ways:

- Sends a large number of unresolvable IP packets to have the receiving device busy with resolving IP addresses until its CPU is overloaded. Unresolvable IP packets refer to IP packets for which ARP cannot find corresponding MAC addresses.
- Sends a large number of ARP packets to overload the CPU of the receiving device.
- Acts as a trusted user or gateway to send ARP packets so the receiving devices obtain incorrect ARP entries.

ARP attack protection tasks at a glance

All ARP attack protection tasks are optional.

- Preventing flood attacks
  - Configuring unresolvable IP attack protection
  - Configuring source MAC-based ARP attack detection
- Preventing user and gateway spoofing attacks
  - Configuring ARP packet source MAC consistency check
  - Configuring ARP active acknowledgement
  - Configuring authorized ARP
  - Configuring ARP attack detection
  - Configuring ARP scanning and fixed ARP
  - Configuring ARP gateway protection
  - Configuring ARP filtering

Configuring unresolvable IP attack protection

About unresolvable IP attack protection

If a device receives a large number of unresolvable IP packets from a host, the following situations can occur:

- The device sends a large number of ARP requests, overloading the target subnets.
- The device keeps trying to resolve the destination IP addresses, overloading its CPU.

To protect the device from such IP attacks, you can configure ARP source suppression. This feature stops resolving packets from an IP address if the number of unresolvable IP packets from the IP address exceeds the upper limit within 5 seconds. The device continues ARP resolution when the interval elapses. This feature is applicable if the attack packets have the same source addresses.

Configuring ARP source suppression

1. Enter system view.

```
system-view
```

2. Enable ARP source suppression.

```
arp source-suppression enable
```

By default, ARP source suppression is disabled.

3. Set the maximum number of unresolvable packets that the device can process per source IP address within 5 seconds.

```
arp source-suppression limit limit-value
```

By default, the maximum number is 10.

Display and maintenance commands for unresolvable IP attack protection

Execute **display** commands in any view.

Task	Command
Display ARP source suppression configuration information.	**display arp source-suppression**

Configuring source MAC-based ARP attack detection

About source MAC-based ARP attack detection

This feature checks the number of ARP packets delivered to the CPU. If the number of packets from the same MAC address within 5 seconds exceeds a threshold, the device generates an ARP attack entry for the MAC address. If the ARP logging feature is enabled, the device handles the attack by using either of the following methods before the ARP attack entry ages out:

- **Monitor**—Only generates log messages.
- **Filter**—Generates log messages and filters out subsequent ARP packets from the MAC address.

To enable the ARP logging feature, use the **arp check log enable** command. For information about the ARP logging feature, see ARP configuration in *Layer 3—IP Services Configuration Guide*.

When an ARP attack entry ages out, ARP packets sourced from the MAC address in the entry can be processed correctly.

Restrictions and guidelines

When you change the handling method from monitor to filter, the configuration takes effect immediately. When you change the handling method from filter to monitor, the device continues filtering packets that match existing attack entries.

You can exclude the MAC addresses of some gateways and servers from this detection. This feature does not inspect ARP packets from those devices even if they are attackers.

Procedure

1. Enter system view.

```
system-view
```

2. Enable source MAC-based ARP attack detection and specify the handling method.

   ```
   arp source-mac { filter | monitor }
   ```

   By default, this feature is disabled.

3. Set the threshold.

   ```
   arp source-mac threshold threshold-value
   ```

   By default, the threshold for source MAC-based ARP attack detection is 30.

4. Set the aging timer for ARP attack entries.

   ```
   arp source-mac aging-time time
   ```

   By default, the lifetime is 300 seconds.

5. (Optional.) Exclude specific MAC addresses from this detection.

   ```
   arp source-mac exclude-mac mac-address&<1-n>
   ```

   By default, no MAC address is excluded.

Display and maintenance commands for source MAC-based ARP attack detection

Execute **display** commands in any view.

Task	Command
Display ARP attack entries detected by source MAC-based ARP attack detection.	**display arp source-mac** { **interface** *interface-type interface-number* | **slot** *slot-number* }

Configuring ARP packet source MAC consistency check

About this task

This feature enables a gateway to filter out ARP packets whose source MAC address in the Ethernet header is different from the sender MAC address in the message body. This feature allows the gateway to learn correct ARP entries.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable ARP packet source MAC address consistency check.

   ```
   arp valid-check enable
   ```

   By default, ARP packet source MAC address consistency check is disabled.

Configuring ARP active acknowledgement

About this task

Use the ARP active acknowledgement feature on gateways to prevent user spoofing.

This feature enables the device to perform active acknowledgement before creating an ARP entry.

- Upon receiving an ARP request that requests the MAC address of the device, the device sends an ARP reply. Then, it sends an ARP request for the sender IP address in the received ARP request to determine whether to create an ARP entry for the sender IP address.
    - If the device receives an ARP reply within the probe interval, it creates the ARP entry.
    - If the device does not receive an ARP reply within the probe interval, it does not create the ARP entry.
- Upon receiving an ARP reply, the device examines whether it was the reply to the request that the device has sent.
    - If it was, the device creates an ARP entry for the sender IP address in the ARP reply.
    - If it was not, the device sends an ARP request for the sender IP address to determine whether to create an ARP entry for the sender IP address.
        - If the device receives an ARP reply within the probe interval, it creates the ARP entry.
        - If the device does not receive an ARP reply within the probe interval, it does not create the ARP entry.

To improve validity and reliability of ARP entries, you can enable ARP active acknowledgement in strict mode. In this mode, the device creates ARP entries only for the IP addresses that the device actively initiates the ARP resolution.

Procedure

1. Enter system view.

   `system-view`

2. Enable the ARP active acknowledgement feature.

   `arp active-ack enable`

   By default, this feature is disabled.

Configuring authorized ARP

About authorized ARP

Authorized ARP entries are generated based on the DHCP clients' address leases on the DHCP server or dynamic client entries on the DHCP relay agent. For more information about DHCP server and DHCP relay agent, see *Layer 3—IP Services Configuration Guide*.

Use this feature to prevent user spoofing and to allow only authorized clients to access network resources.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   Supported interface types include Layer 3 Ethernet interface, Layer 3 Ethernet subinterface, Layer 3 aggregate interface, Layer 3 aggregate subinterface, and VLAN interface.

3. Enable authorized ARP on the interface.

   `arp authorized enable`

   By default, authorized ARP is disabled.

Configuring ARP attack detection

About ARP attack detection

ARP attack detection enables access devices to block ARP packets from unauthorized clients to prevent user spoofing and gateway spoofing attacks.

ARP attack detection provides the following features:

- User validity check.
- ARP packet validity check.
- ARP restricted forwarding.
- ARP attack detection logging.

If both ARP packet validity check and user validity check are enabled, the former one applies first, and then the latter applies.

Do not configure ARP attack detection together with ARP snooping. Otherwise, ARP snooping entries cannot be generated.

Configuring user validity check

About this task

User validity check does not check ARP packets received on ARP trusted interfaces. This feature compares the sender IP and sender MAC in the ARP packet received on an ARP untrusted interface with the matching criteria in the following order:

1. User validity check rules.
   - If a match is found, the device processes the ARP packet according to the rule.
   - If no match is found or no user validity check rule is configured, proceeds to step 2.
2. Static IPSG bindings.
   - If a match is found, the device determines that the ARP packet is valid. Then, the device forwards the packet by searching for an entry that contains the target IP address.
     - If a match is found and the receiving interface is different from the interface in the entry with a matching sender IP address, the device performs Layer 3 forwarding.
     - If a match is found but the receiving interface is the same as the interface in the entry with a matching sender IP address, the device performs Layer 2 forwarding.
     - If no match is found, the device performs Layer 2 forwarding.
   - If no match is found, the device discards the ARP packet.

Static IP source guard bindings are created by using the **`ip source binding`** command. For more information, see "Configuring IP source guard."

Restrictions and guidelines

When you configure user validity check, make sure one or more of the following items are configured:

- User validity check rules.
- Static IP source guard bindings.

If neither of the items is configured, all incoming ARP packets on ARP untrusted interfaces are discarded.

Specify an IP address, a MAC address, and a VLAN where ARP attack detection is enabled for an IP source guard binding. Otherwise, no ARP packets can match the IP source guard binding.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure a user validity check rule.

   arp detection rule *rule-id* { **deny** | **permit** } **ip** { *ip-address* [*mask*] | **any** } **mac** { *mac-address* [*mask*] | **any** } [**vlan** *vlan-id*]

   By default, no user validity check rules are configured.

3. Enter VLAN view.

   vlan *vlan-id*

4. Enable ARP attack detection.

   arp detection enable

   By default, ARP attack detection is disabled. The device does not perform user validity check.

5. (Optional.) Configure an interface that does not require ARP user validity check as a trusted interface.

   a. Return to system view.

      quit

   b. Enter interface view.

      interface *interface-type interface-number*

      Supported interface types include Layer 2 Ethernet interface and Layer 2 aggregate interface.

   c. Configure the interface as a trusted interface excluded from ARP attack detection.

      arp detection trust

      By default, an interface is untrusted.

Configuring ARP packet validity check

About this task

ARP packet validity check does not check ARP packets received on ARP trusted interfaces. To check ARP packets received on untrusted interfaces, you can specify the following objects to be checked:

- **src-mac**—Checks whether the sender MAC address in the message body is identical to the source MAC address in the Ethernet header. If they are identical, the packet is forwarded. Otherwise, the packet is discarded.
- **dst-mac**—Checks the target MAC address of ARP replies. If the target MAC address is all-zero, all-one, or inconsistent with the destination MAC address in the Ethernet header, the packet is considered invalid and discarded.
- **ip**—Checks the sender and target IP addresses of ARP replies, and the sender IP address of ARP requests. All-one or multicast IP addresses are considered invalid and the corresponding packets are discarded.

Prerequisites

Before you configure ARP packet validity check, you must first configure user validity check. For more information about user validity check configuration, see "Configuring user validity check."

Procedure

1. Enter system view.

   system-view

2. Enter VLAN view.

```
vlan vlan-id
```

3. Enable ARP attack detection.

   ```
   arp detection enable
   ```

   By default, ARP attack detection is disabled. The device does not perform ARP packet validity check.

4. Enable ARP packet validity check.
   a. Return to system view.

      ```
      quit
      ```

   b. Enable ARP packet validity check and specify the objects to be checked.

      ```
      arp detection validate { dst-mac | ip | src-mac } *
      ```

      By default, ARP packet validity check is disabled.

5. (Optional.) Configure the interface that does not require ARP packet validity check as a trusted interface.
   a. Enter interface view.

      ```
      interface interface-type interface-number
      ```

      Supported interface types include Layer 2 Ethernet interface and Layer 2 aggregate interface.

   b. Configure the interface as a trusted interface excluded from ARP attack detection.

      ```
      arp detection trust
      ```

      By default, an interface is untrusted.

Configuring ARP restricted forwarding

About this task

ARP restricted forwarding does not take effect on ARP packets received on ARP trusted interfaces and forwards the ARP packets correctly. This feature controls the forwarding of ARP packets that are received on untrusted interfaces and have passed user validity check as follows:

- If the packets are ARP requests, they are forwarded through the trusted interface.
- If the packets are ARP replies, they are forwarded according to their destination MAC address. If no match is found in the MAC address table, they are forwarded through the trusted interface.

Restrictions and guidelines

ARP restricted forwarding does not apply to ARP packets that use multiport destination MAC addresses.

Prerequisites

Configure user validity check before you configure ARP restricted forwarding. For information about user validity check configuration, see "Configuring user validity check."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter VLAN view.

   ```
   vlan vlan-id
   ```

3. Enable ARP restricted forwarding.

   ```
   arp restricted-forwarding enable
   ```

   By default, ARP restricted forwarding is disabled.

Display and maintenance commands for ARP attack detection

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the VLANs enabled with ARP attack detection.	**display arp detection**
Display statistics for packets dropped by ARP attack detection.	**display arp detection statistics** [**interface** *interface-type interface-number*]
Clear statistics for packets dropped by ARP attack detection.	**reset arp detection statistics** [**interface** *interface-type interface-number*]

Configuring ARP scanning and fixed ARP

About this task

ARP scanning is typically used together with the fixed ARP feature in small-scale and stable networks.

ARP scanning automatically creates ARP entries for devices in an address range. The device performs ARP scanning in the following steps:

1. Sends ARP requests for each IP address in the address range.
2. Obtains their MAC addresses through received ARP replies.
3. Creates dynamic ARP entries.

Fixed ARP converts existing dynamic ARP entries (including those generated through ARP scanning) to static ARP entries. These static ARP entries are of the same attributes as the ARP entries that are manually configured. This feature prevents ARP entries from being modified by attackers.

Restrictions and guidelines

IP addresses in existing ARP entries are not scanned.

Due to the limit on the total number of static ARP entries, some dynamic ARP entries might fail the conversion.

The **arp fixup** command is a one-time operation. You can use this command again to convert the dynamic ARP entries learned later to static.

To delete a static ARP entry converted from a dynamic one, use the **undo arp** *ip-address* [*vpn-instance-name*] command. You can also use the **reset arp all** command to delete all ARP entries or the **reset arp static** command to delete all static ARP entries.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Trigger an ARP scanning.
   arp scan [*start-ip-address* **to** *end-ip-address*]

△ **CAUTION:**

ARP scanning will take some time and occupy a lot of system and network resources. To stop an ongoing scan, press **Ctrl** + **C**. Dynamic ARP entries are created based on ARP replies received before the scan is terminated.

4. Return to system view.
   `quit`
5. Convert existing dynamic ARP entries to static ARP entries.
   `arp fixup`

Configuring ARP gateway protection

About ARP gateway protection

Configure this feature on interfaces not connected with a gateway to prevent gateway spoofing attacks.

When such an interface receives an ARP packet, it checks whether the sender IP address in the packet is consistent with that of any protected gateway. If yes, it discards the packet. If not, it handles the packet correctly.

Restrictions and guidelines

You can enable ARP gateway protection for a maximum of eight gateways on an interface.

Do not configure both the **`arp filter source`** and **`arp filter binding`** commands on an interface.

If ARP gateway protection works with ARP attack detection and ARP snooping, ARP gateway protection applies first.

Procedure

1. Enter system view.
   `system-view`
2. Enter interface view.
   `interface` *`interface-type interface-number`*
   Supported interface types include Layer 2 Ethernet interface and Layer 2 aggregate interface.
3. Enable ARP gateway protection for the specified gateway.
   `arp filter source` *`ip-address`*
   By default, ARP gateway protection is disabled.

Configuring ARP filtering

ARP filtering

The ARP filtering feature can prevent gateway spoofing and user spoofing attacks.

An interface enabled with this feature checks the sender IP and MAC addresses in a received ARP packet against permitted entries. If a match is found, the packet is handled correctly. If not, the packet is discarded.

Restrictions and guidelines

You can configure a maximum of eight permitted entries on an interface.

Do not configure both the **`arp filter source`** and **`arp filter binding`** commands on an interface.

If ARP filtering works with ARP attack detection and ARP snooping, ARP filtering applies first.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   Supported interface types include Ethernet interface and Layer 2 aggregate interface.

3. Enable ARP filtering and configure a permitted entry.

   `arp filter binding` *`ip-address mac-address`*

   By default, ARP filtering is disabled.

Contents

Configuring ND attack defense

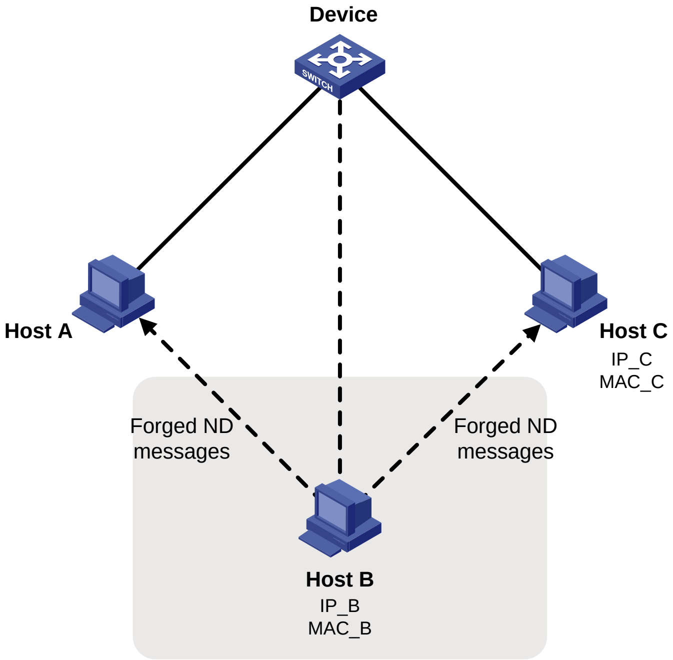
About ND attack defense

IPv6 Neighbor Discovery (ND) attack defense is able to identify forged ND messages to prevent ND attacks.

The IPv6 ND protocol does not provide any security mechanisms and is vulnerable to network attacks. As shown in Figure 1, an attacker can send the following forged ICMPv6 messages to perform ND attacks:

- Forged NS/NA/RS messages with an IPv6 address of a victim host. The gateway and other hosts update the ND entry for the victim with incorrect address information. As a result, all packets intended for the victim are sent to the attacking terminal.
- Forged RA messages with the IPv6 address of a victim gateway. As a result, all hosts attached to the victim gateway maintain incorrect IPv6 configuration parameters and ND entries.

Figure 1 ND attack diagram



Configuring source MAC-based ND attack detection

About source MAC-based ND attack detection

Source MAC-based ND attack detection checks the number of ND messages delivered to the CPU on a per source MAC basis. If the number of messages from the same MAC address within 5 seconds exceeds the threshold, the device generates an ND attack entry for the MAC address. The processing of the ND messages matching this entry depends on the detection mode. With ND logging enabled (by using the **`ipv6 nd check log enable`** command), source MAC-based ND attack detection processes the messages as follows:

- **Filter mode**—Filters out subsequent ND messages sent from the MAC address, and generates log messages.
- **Monitor mode**—Only generates log messages.

The device uses the entry aging time (fixed at 300 seconds) and the threshold to calculate a value:

The calculated value = (threshold/5) × 300

The device monitors the number of dropped packets for an entry. When the entry aging time is reached, it compares the number with the calculated value and takes actions accordingly:

- If the number of dropped packets is higher than or equal to the calculated value, the device resets the aging time for the entry.
- If the number of dropped packets is lower than the calculated value, the system deletes the entry and marks MAC address in the entry as a common MAC address.

Restrictions and guidelines

When you change the detection mode from monitor to filter, the filter mode takes effect immediately.

When you change the detection mode from filter to monitor, the device continues filtering messages that match existing attack entries.

Procedure

1. Enter system view.
   `system-view`
2. Enable source MAC-based ND attack detection and set the detection mode.
   `ipv6 nd source-mac { filter | monitor }`
   By default, source MAC-based ND attack detection is disabled.
3. Set the threshold for source MAC-based ND attack detection.
   `ipv6 nd source-mac threshold` *`threshold-value`*
   The default setting is 30.
4. Enable the ND logging feature.
   `ipv6 nd check log enable`
   By default, the ND logging feature is disabled.

Display and maintenance commands for source MAC-based ND attack detection

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display the configuration of source MAC-based ND attack detection.	**`display ipv6 nd source-mac configuration`**

Display source MAC-based ND attack detection entries.	**`display ipv6 nd source-mac interface`** *`interface-type interface-number`* `[` **`slot`** *`slot-number`* `] [` **`verbose`** `]`
	`display ipv6 nd source-mac` `{` **`mac`** *`mac-address`* `|` **`vlan`** *`vlan-id`* `}` **`slot`** *`slot-number`* `[` **`verbose`** `]`

Task	Command
	display ipv6 nd source-mac slot *slot-number* [**count** | **verbose**]
Delete source MAC-based ND attack detection entries.	**reset ipv6 nd source-mac** [**interface** *interface-type interface-number* | **mac** *mac-address* | **vlan** *vlan-id*] [**slot** *slot-number*]

Configuring interface-based ND attack suppression

About interface-based ND attack suppression

This feature rate limits ND request on each Layer 3 interface to prevent ND spoofing attacks. It monitors the number of ND requests that each Layer 3 interface received within 5 seconds. If the number on an interface exceeds the threshold, the device creates an ND attack suppression entry for the interface. During the suppression period (fixed at 300 seconds), the device drops ND messages received on this interface.

When the suppression time expires, the system examines the number of dropped ND messages on the interface within the suppression time:

- If the number is higher than or equal to the calculated value, the device resets the suppression time for the entry and continues ND suppression on the interface.

  The calculated value = (threshold/5) × 300
- If the number is lower than the calculated value, the device deletes the suppression entry.

Restrictions and guidelines

As a best practice, enable this feature on the gateway.

Procedure

1. Enter system view.

   system-view
2. Enable interface-based ND attack suppression.

   ipv6 nd attack-suppression enable per-interface

   By default, interface-based ND attack suppression is disabled.
3. Set the threshold for triggering ND attack suppression.

   ipv6 nd attack-suppression threshold *threshold-value*

   By default, the threshold for triggering ND attack suppression is 1000.

Display and maintenance commands for interface-based ND attack suppression

Execute **display** commands in any view.

Task	Command
Display the configuration of interface-based ND attack suppression.	**display ipv6 nd attack-suppression configuration**
Display interface-based ND attack suppression entries.	**display ipv6 nd attack-suppression per-interface slot** *slot-number* [**count** | **verbose**]
Display interface-based ND attack suppression entries on an interface.	**display ipv6 nd attack-suppression per-interface interface** *interface-type interface-number* [**verbose**]
Delete interface-based ND attack suppression entries.	**reset ipv6 nd attack-suppression per-interface** [**interface** *interface-type interface-number*] [**slot** *slot-number*]
Clear statistics for ND messages dropped by interface-based ND attack suppression.	**reset ipv6 nd attack-suppression per-interface statistics** [**interface** *interface-type interface-number*] [**slot** *slot-number*]

Enabling source MAC consistency check for ND messages

About this task

The source MAC consistency check feature is typically configured on gateways to prevent ND attacks.

This feature checks the source MAC address and the source link-layer address for consistency for each arriving ND message.

- If the source MAC address and the source link-layer address are not the same, the device drops the packet.
- If the addresses are the same, the device continues learning ND entries.

The ND logging feature logs source MAC inconsistency events, and it sends the log messages to the information center. The information center can then output log messages from different source modules to different destinations. For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enable source MAC consistency check for ND messages.

```
ipv6 nd mac-check enable
```

By default, source MAC consistency check is disabled for ND messages.

3. (Optional.) Enable the ND logging feature.

```
ipv6 nd check log enable
```

By default, the ND logging feature is disabled.

As a best practice, disable the ND logging feature to avoid excessive ND logs.

Contents

Configuring uRPF

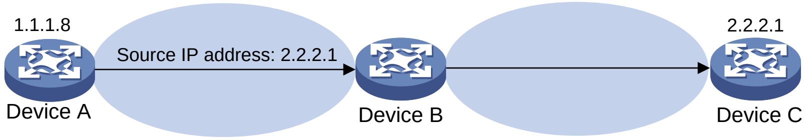
About uRPF

Unicast Reverse Path Forwarding (uRPF) protects a network against source address spoofing attacks, such as DoS and DDoS attacks.

uRPF application scenario

Attackers send packets with a forged source address to access a system that uses IPv4-based authentication, in the name of authorized users or even the administrator. Even if the attackers or other hosts cannot receive any response packets, the attacks are still disruptive to the attacked target.

Figure 1 Source address spoofing attack



As shown in Figure 1, an attacker on Device A sends the server (Device B) requests with a forged source IP address 2.2.2.1 at a high rate. Device B sends response packets to IP address 2.2.2.1 (Device C). Consequently, both Device B and Device C are attacked. If the administrator disconnects Device C by mistake, the network service is interrupted.

Attackers can also send packets with different forged source addresses or attack multiple servers simultaneously to block connections or even break down the network.

uRPF can prevent these source address spoofing attacks. It checks whether an interface that receives a packet is the output interface of the FIB entry that matches the source address of the packet. If not, uRPF considers it a spoofing attack and discards the packet.

uRPF check modes

uRPF supports strict and loose modes.

Strict uRPF check

To pass strict uRPF check, the source address of a packet and the receiving interface must match the destination address and output interface of a FIB entry. In some scenarios (for example, asymmetrical routing), strict uRPF might discard valid packets.

Strict uRPF is often deployed between a PE and a CE.

Loose uRPF check

To pass loose uRPF check, the source address of a packet must match the destination address of a FIB entry. Loose uRPF can avoid discarding valid packets, but might let go attack packets.

Loose uRPF is often deployed between ISPs, especially in asymmetrical routing.

uRPF extended functions

Link layer check

Strict uRPF check can further perform link layer check on a packet. It uses the next hop address in the matching FIB entry to look up the ARP table for a matching entry. If the source MAC address of the packet matches the MAC address in the matching ARP entry, the packet passes strict uRPF check. Link layer check is applicable to ISP devices where a Layer 3 Ethernet interface connects a large number of PCs.

Loose uRPF does not support link layer check.

Using the default route in uRPF check

When a default route exists, all packets that fail to match a specific FIB entry match the default route during uRPF check and thus are permitted to pass. To avoid this situation, you can disable uRPF from using any default route to discard such packets. If you allow using the default route (set by using **`allow-default-route`**), uRPF permits packets that only match the default route.

By default, uRPF discards packets that can only match a default route.

Typically, you do not need to configure the **`allow-default-route`** keyword on a PE device because it has no default route pointing to the CE. If you enable uRPF on a security zone where the CE interface resides and the security zone has a default route pointing to the PE, specify the **`allow-default-route`** keyword.

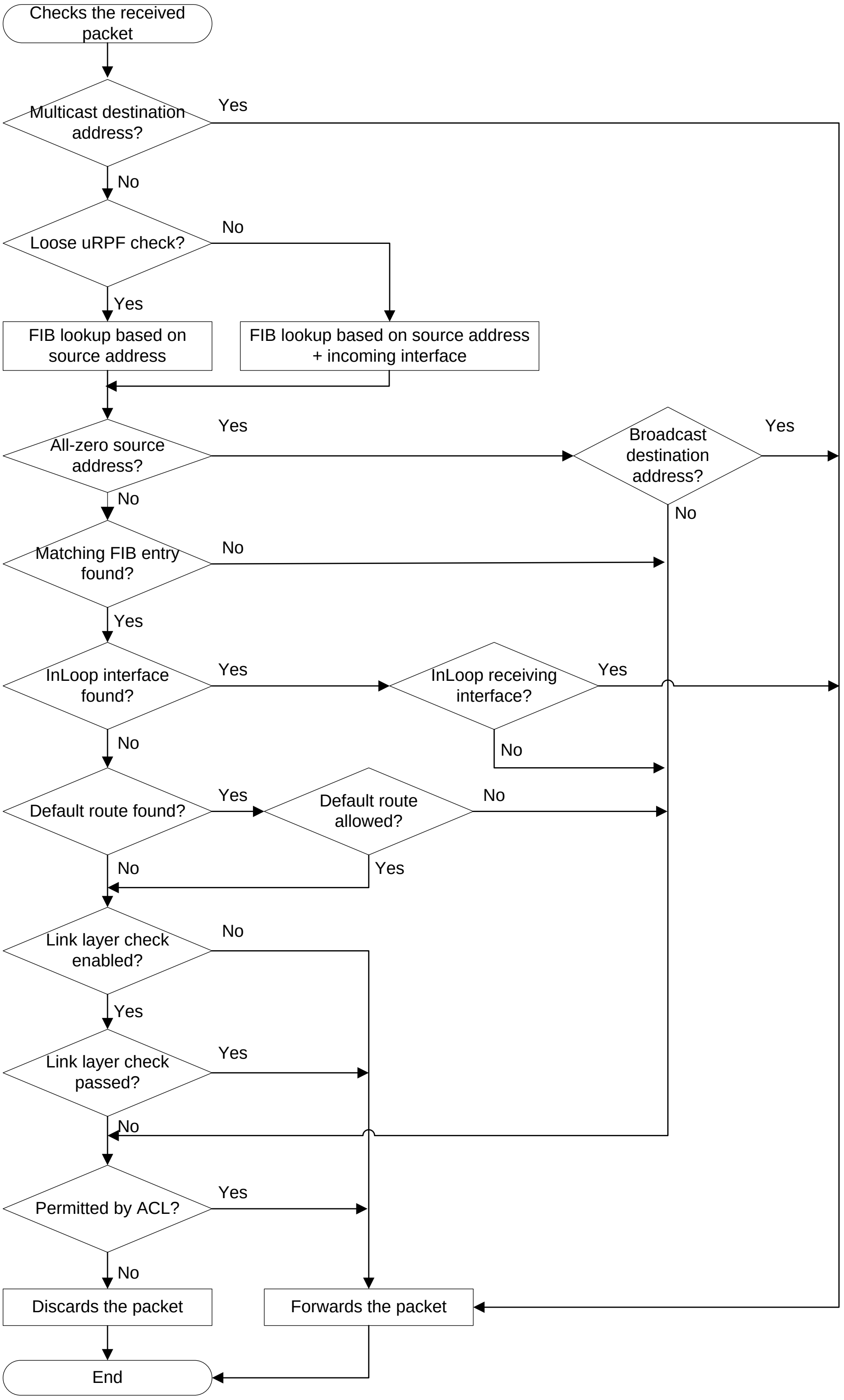
Using an ACL for uRPF check exemption

To identify specific packets as valid packets, you can use an ACL to match these packets. Even if the packets do not pass uRPF check, they are still forwarded.

uRPF operation

Figure 2 shows how uRPF works.

Figure 2 uRPF work flow


Checks the received packet
Multicast destination address?
Yes
No
Loose uRPF check?
No
Yes
FIB lookup based on source address
FIB lookup based on source address + incoming interface
All-zero source address?
Yes
Broadcast destination address?
Yes
No
No
Matching FIB entry found?
No
Yes
InLoop interface found?
Yes
InLoop receiving interface?
Yes
No
No
Default route found?
Yes
Default route allowed?
No
No
Yes
Link layer check enabled?
No
Yes
Link layer check passed?
Yes
No
Permitted by ACL?
Yes
No
Discards the packet
Forwards the packet
End

1. uRPF checks whether the received packet carries a multicast destination address:
   - If yes, uRPF permits the packet.
   - If no, uRPF proceeds to step 2.
2. uRPF checks whether the uRPF check mode is loose:

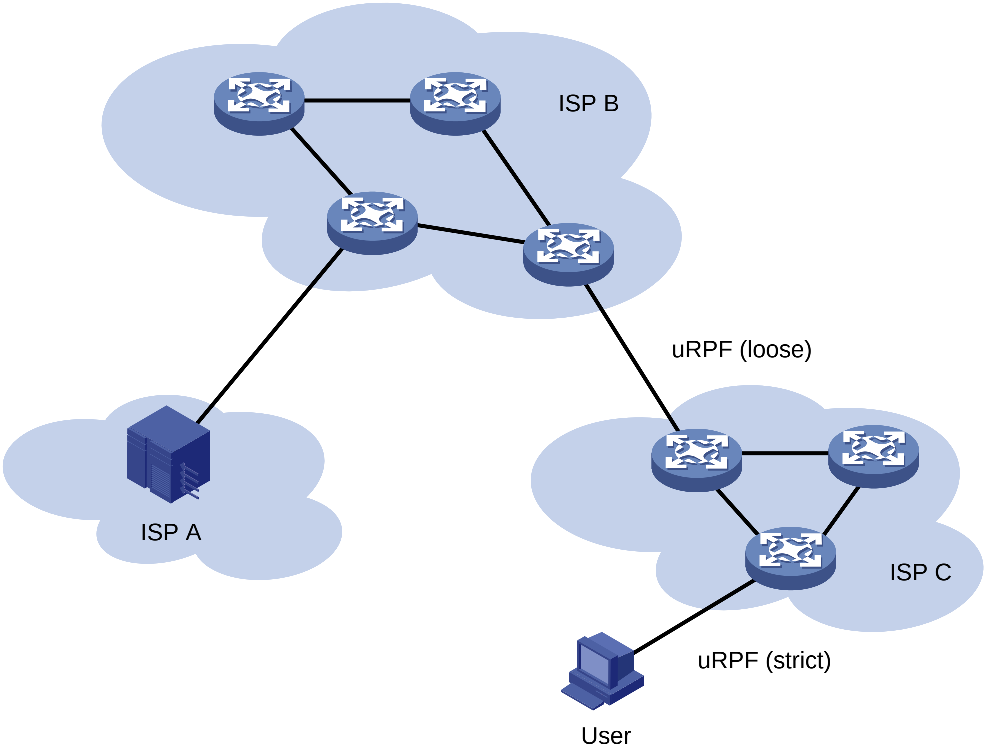
    - If yes, uRPF performs FIB lookup based on the source IP address and then proceeds to step 3.
    - If no, uRPF performs FIB lookup based on the source IP address and the receiving interface and then proceeds to step 3.
3. uRPF checks whether the source IP address is an all-zero address:
    - If yes, uRPF checks whether the destination address of the packet is a broadcast address.
        - If yes, uRPF permits the packet.
        - If no, uRPF proceeds to step 8.
    - If no, uRPF proceeds to step 4.
4. uRPF checks whether the source address matches a unicast route:
    - If yes, uRPF proceeds to step 5.
    - If no, uRPF proceeds to step 8.
5. uRPF checks whether the matching route is to the host itself (whether the output interface of the matching route is an InLoop interface):
    - If yes, uRPF checks whether the receiving interface of the packet is an InLoop interface.
        - If yes, uRPF permits the packet.
        - If no, uRPF proceeds to step 8.
    - If no, uRPF proceeds to step 6.
6. uRPF checks whether the matching route is a default route:
    - If yes, uRPF checks whether the **`allow-default-route`** keyword is configured to allow using the default route.
        - If yes, uPRF proceeds to step 7.
        - If no, uPRF proceeds to step 8.
    - If no, uRPF proceeds to step 7.
7. uRPF checks whether the **`link-check`** keyword is configured for link layer check:
    - If no, uRPF permits the packet.
    - If yes, uRPF uses the next-hop address of the FIB entry to look up the ARP table for a matching entry. Then it checks whether the MAC address of the matching ARP entry is identical with the source MAC address of the packet.
        - If yes, uRPF permits the packet.
        - If no, uRPF proceeds to step 8.
8. uRPF checks whether the packet is permitted by the ACL:
    - If yes, the packet is forwarded (such a packet is displayed in the uRPF information as a "suppressed drop").
    - If no, the packet is discarded.

Network application

As shown in Figure 3, strict uRPF check is configured between an ISP network and a customer network. Loose uRPF check is configured between ISPs.

For special packets or users, you can configure ACLs.

Figure 3 Network diagram



Restrictions and guidelines: uRPF configuration

Do not configure the **`allow-default-route`** keyword for loose uRPF check. Otherwise, uRPF might fail to work.

Do not use strict uRPF if ECMP routing is available in the network. Service packets that travel along ECMP routes cannot pass the strict uRPF check and will be dropped.

Enabling uRPF for a security zone

Restrictions and guidelines

uRPF enabled for a security zone takes effect on all interfaces in the security zone.

Procedure

1. Enter system view.

   `system-view`

2. Enter security zone view.

   `security-zone name` *`zone-name`*

3. Enable uRPF.

   `ip urpf` { **`loose`** [**`allow-default-route`**] [**`acl`** *`acl-number`*] | **`strict`** [**`allow-default-route`**] [**`acl`** *`acl-number`*] [**`link-check`**] }

   By default, uRPF is disabled.

Display and maintenance commands for uRPF

Execute **display** commands in any view.

Task	Command
Display uRPF configuration.	**display ip urpf** [**security-zone** *zone-name*] [**slot** *slot-number*]
Display uRPF statistics for a security zone.	**display ip urpf statistics security-zone** *zone-name* [**slot** *slot-number*]
Clear uRPF statistics for a security zone.	**reset ip urpf statistics security-zone** *zone-name*

Configuring IPv6 uRPF

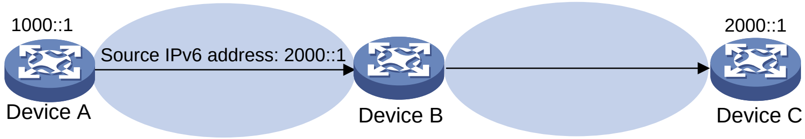
About IPv6 uRPF

IPv6 Unicast Reverse Path Forwarding (uRPF) protects a network against source address spoofing attacks, such as DoS and DDoS attacks.

IPv6 uRPF application scenario

Attackers send packets with a forged source address to access a system that uses IPv6-based authentication, in the name of authorized users or even the administrator. Even if the attackers or other hosts cannot receive any response packets, the attacks are still disruptive to the attacked target.

Figure 4 Source address spoofing attack



As shown in Figure 4, an attacker on Device A sends the server (Device B) requests with a forged source IPv6 address 2000::1 at a high rate. Device B sends response packets to IPv6 address 2000::1 (Device C). Consequently, both Device B and Device C are attacked. If the administrator disconnects Device C by mistake, the network service is interrupted.

Attackers can also send packets with different forged source addresses or attack multiple servers simultaneously to block connections or even break down the network.

IPv6 uRPF can prevent these source address spoofing attacks. It checks whether an interface that receives a packet is the output interface of the FIB entry that matches the source address of the packet. If not, IPv6 uRPF considers it a spoofing attack and discards the packet.

IPv6 uRPF check modes

IPv6 uRPF supports strict and loose check modes.

Strict IPv6 uRPF check

To pass strict IPv6 uRPF check, the source address of a packet and the receiving interface must match the destination address and output interface of an IPv6 FIB entry. In some scenarios (for example, asymmetrical routing), strict IPv6 uRPF might discard valid packets.

Strict IPv6 uRPF is often deployed between a PE and a CE.

Loose IPv6 uRPF check

To pass loose IPv6 uRPF check, the source address of a packet must match the destination address of an IPv6 FIB entry. Loose IPv6 uRPF can avoid discarding valid packets, but might let go attack packets.

Loose IPv6 uRPF is often deployed between ISPs, especially in asymmetrical routing.

IPv6 uRPF extended functions

Using the default route in IPv6 uRPF check

When a default route exists, all packets that fail to match a specific IPv6 FIB entry match the default route during IPv6 uRPF check and thus are permitted to pass. If you allow using the default route (by using **`allow-default-route`**), IPv6 uRPF permits packets that only match the default route.

By default, IPv6 uRPF discards packets that can only match a default route.

Typically, you do not need to configure the **`allow-default-route`** keyword on a PE device because it has no default route pointing to the CE device. If you enable uRPF on a security zone where the CE interface resides and the security zone has a default route pointing to the PE, specify the **`allow-default-route`** keyword.

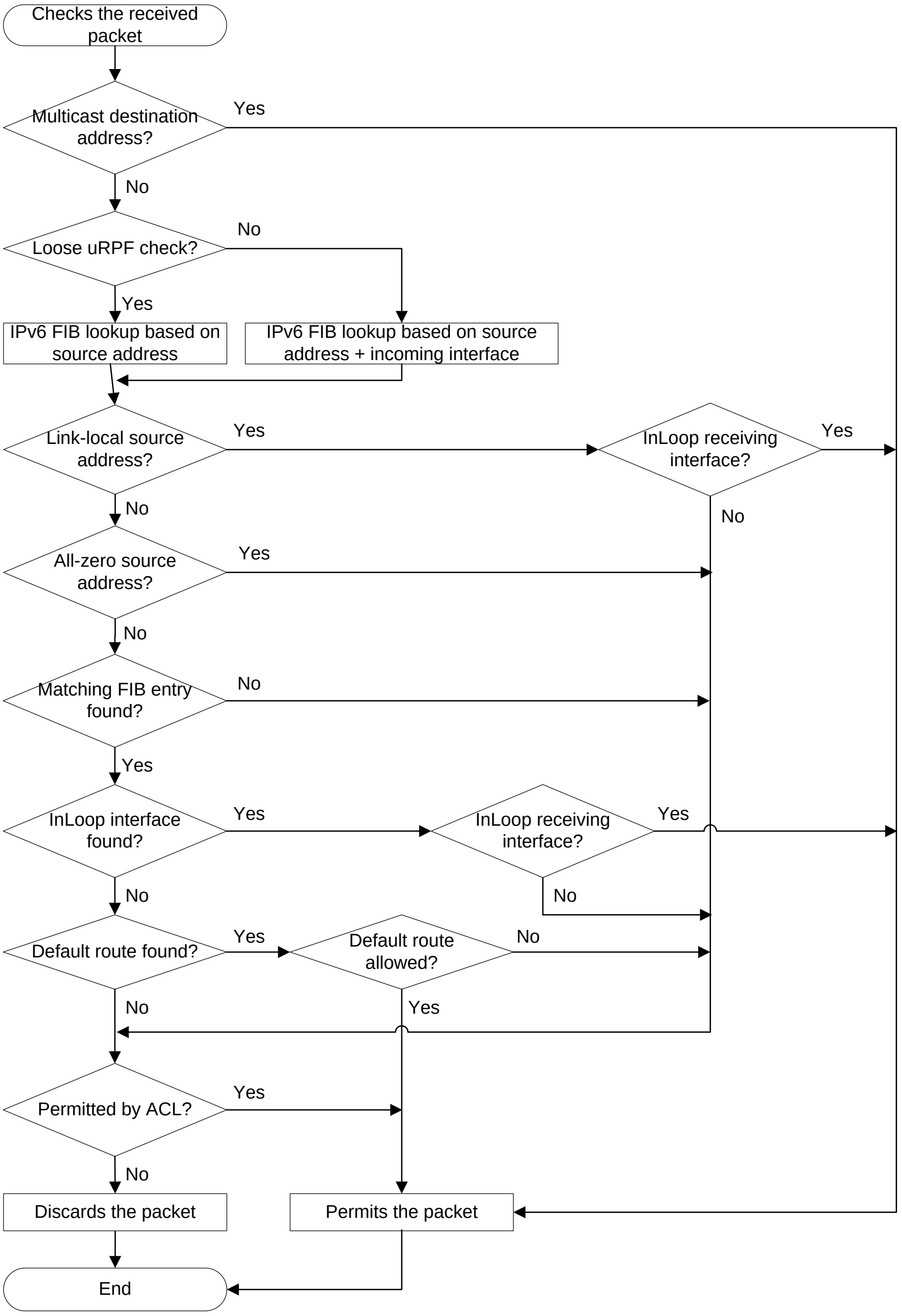
Using an ACL for IPv6 uRPF check exemption

To identify specific packets as valid packets, you can use an IPv6 ACL to match these packets. Even if the packets do not pass IPv6 uRPF check, they are still forwarded.

IPv6 uRPF operation

Figure 5 shows how IPv6 uRPF works.

Figure 5 IPv6 uRPF work flow


Checks the received packet
Multicast destination address?
Yes
No
Loose uRPF check?
No
Yes
IPv6 FIB lookup based on source address
IPv6 FIB lookup based on source address + incoming interface
Link-local source address?
Yes
InLoop receiving interface?
Yes
No
No
All-zero source address?
Yes
No
Matching FIB entry found?
No
Yes
InLoop interface found?
Yes
InLoop receiving interface?
Yes
No
No
Default route found?
Yes
Default route allowed?
No
No
Yes
Permitted by ACL?
Yes
No
Discards the packet
Permits the packet
End

1. IPv6 uRPF checks whether the received packet carries a multicast destination address:
   - If yes, IPv6 uRPF permits the packet.
   - If no, IPv6 uRPF proceeds to step 2.
2. IPv6 uRPF checks whether the IPv6 uRPF check mode is loose:
   - If yes, IPv6 uRPF performs IPv6 FIB lookup based on the source IPv6 address and then proceeds to step 3.
   - If no, IPv6 uRPF performs IPv6 FIB lookup based on the source IPv6 address and the receiving interface and then proceeds to step 3.
3. IPv6 uRPF checks whether the source address is a link-local address:
   - If yes, IPv6 uRPF checks whether the receiving interface is a InLoop interface.

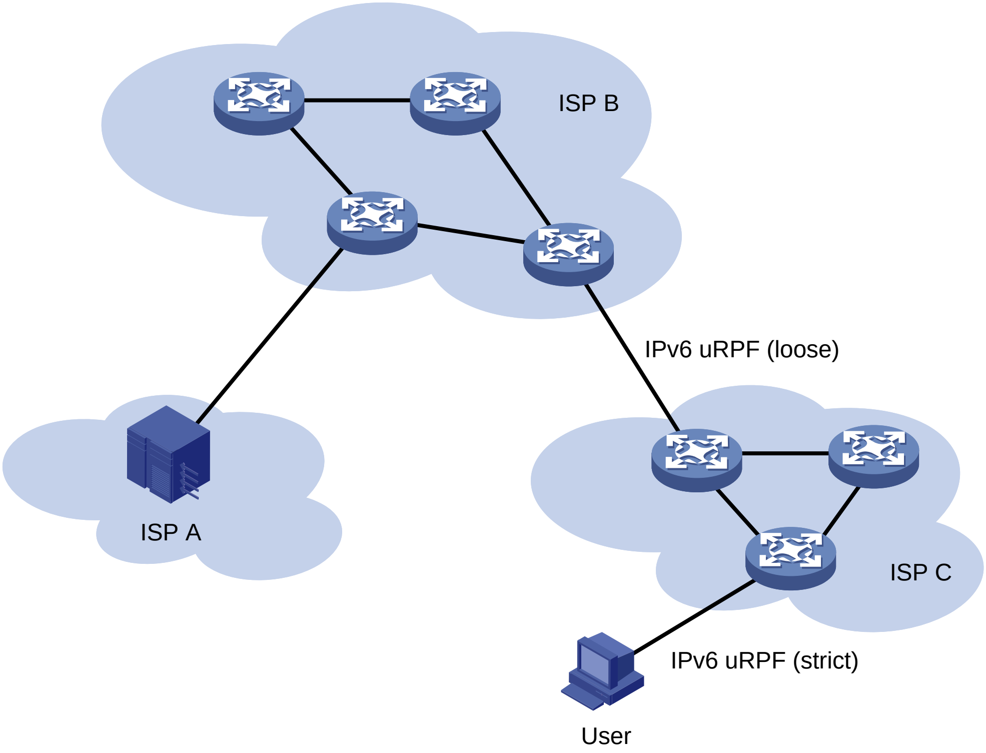
    - If yes, IPv6 uRPF permits the packet.
    - If not, IPv6 uRPF proceeds to step 4.
  - If no, IPv6 uRPF proceeds to step 8.
4. IPv6 uRPF checks whether the source address is an all-zero address:
   - If yes, IPv6 uRPF proceeds to step 8.
   - If no, IPv6 uRPF proceeds to step 5.
5. IPv6 uRPF checks whether the source address matches a unicast route:
   - If yes, IPv6 uRPF proceeds to step 6.
   - If no, IPv6 uRPF proceeds to step 8.
6. IPv6 uRPF checks whether the matching route is to the host itself (whether the output interface of the matching route is an InLoop interface):
   - If yes, IPv6 uRPF checks whether the receiving interface of the packet is an InLoop interface.
     - If yes, IPv6 uRPF permits the packet.
     - If no, IPv6 uRPF proceeds to step 8.
   - If no, IPv6 uRPF proceeds to step 7.
7. IPv6 uRPF checks whether the matching route is a default route:
   - If yes, IPv6 uRPF checks whether the **`allow-default-route`** keyword is configured to allow using the default route.
     - If yes, the packet is forwarded.
     - If no, IPv6 uRPF proceeds to step 8.
   - If no, the packet is forwarded.
8. IPv6 uRPF checks whether the packet is permitted by the IPv6 ACL:
   - If yes, the packet is forwarded (such a packet is displayed in the uRPF information as a "suppressed drop").
   - If no, the packet is discarded.

Network application

As shown in Figure 6, strict IPv6 uRPF check is configured between an ISP network and a customer network. Loose IPv6 uRPF check is configured between ISPs.

For special packets or users, you can configure IPv6 ACLs.

Figure 6 Network diagram



Restrictions and guidelines: IPv6 uRPF configuration

Do not configure the **`allow-default-route`** keyword for loose IPv6 uRPF check. Otherwise, IPv6 uRPF might fail to work.

Do not use strict IPv6 uRPF if ECMP routing is available in the network. Service packets that travel along ECMP routes cannot pass the strict uRPF check and will be dropped.

Enabling IPv6 uRPF for a security zone

Restrictions and guidelines

IPv6 uRPF enabled for a security zone takes effect on all interfaces in the security zone.

Procedure

1. Enter system view.
   `system-view`
2. Enter security zone view.
   `security-zone name` *`zone-name`*
3. Enable IPv6 uRPF.
   `ipv6 urpf` { **`loose`** | **`strict`** } [**`allow-default-route`**] [**`acl`** *`acl-number`*]
   By default, IPv6 uRPF is disabled.

Display and maintenance commands for IPv6 uRPF

Execute **display** commands in any view.

Task	Command
Display IPv6 uRPF configuration.	**display ipv6 urpf** [**security-zone** *zone-name*] [**slot** *slot-number*]
Display IPv6 uRPF statistics for a security zone.	**display ipv6 urpf statistics security-zone** *zone-name* [**slot** *slot-number*]
Clear IPv6 uRPF statistics for a security zone.	**reset ipv6 urpf statistics security-zone** *zone-name*

Contents

Configuring IP-MAC binding

About IP-MAC binding

The device prevents user spoofing attacks by using an IP-MAC binding table to filter out illegitimate packets with forged source IP addresses or MAC addresses.

Operating mechanism

The IP-MAC binding table contains binding entries that bind IP addresses and MAC addresses. The device uses the binding entries to match an incoming packet.

As shown in Figure 1, all hosts communicate with the IP network through the device. When the device receives a packet, it compares the source IP address and source MAC address in the packet with the IP-MAC binding entries. Table 1 describes the way the device processes the packet based on the match result.

Figure 1 IP-MAC binding application scenario

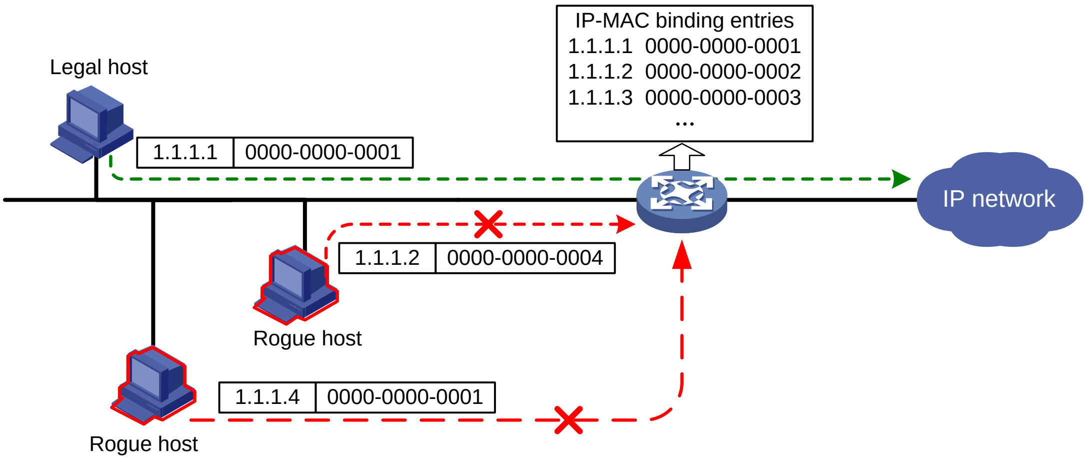


Table 1 Processing of a packet based on the match result

Match result	Processing of the packet
The packet source IP address and source MAC address match the same IP-MAC binding entry.	Permits the packet.
Only the source IP address or source MAC address matches a binding entry.	Drops the packet.
The source IP address and source MAC address match two different binding entries.	Drops the packet.
Both the source IP address and the source MAC address of a packet match no IP-MAC binding entry.	Processes the packet based on the default action. By default, the device permits all packets that do not match any binding entries. You can use the **`ip-mac binding no-match action deny`** command to set the default action to deny.

IP-MAC binding entry creation

An IP-MAC binding entry binds an IP address to a MAC address. You can manually create IP-MAC binding entries one by one or generate them in bulk. All binding entries are globally effective.

Manual creation of IP-MAC binding entries

This method is applicable only to networks that do not contain many hosts and in which the hosts are statically assigned IP addresses.

Bulk generation of IP-MAC binding entries

This method is applicable to networks that contain many hosts.

This method allows a device to generate IPv4-MAC binding entries based on ARP entries and create IPv6-MAC binding entries based on ND entries on an interface.

The device generates an IP-MAC binding entry based on an ARP or ND entry as follows:

- If neither the IP address nor the MAC address in the ARP/ND entry exists in the binding table, the device generates a new binding entry. In this situation, the IP address and the MAC address are uniquely bound to each other.
- If the MAC address in the ARP/ND entry exists in the binding table but the IP address does not, the device generates a new binding entry. In this situation, the MAC address is bound to multiple IP addresses.
- If the IP address in the ARP/ND entry exists in the binding table, the device will not generate a new binding entry. This is because an IP address can be bound to only one MAC address.

IP-MAC binding entries generated based on ARP and ND entries are static. Therefore, the binding entries are not updated when the relevant ARP or ND entries change.

Restrictions and guidelines: IP-MAC binding configuration

IP-MAC binding entries are static. Therefore, the IP-MAC binding feature is applicable only to networks where all users are statically assigned IP addresses. Using this feature in a network where all users' IP addresses are dynamically assigned through DHCP might cause communication failure.

A MAC address can be bound to multiple IP addresses. To bind a MAC address in a binding entry to another IP address, use the MAC address and new IP address to create a new binding entry. You can choose to delete the existing binding entry or retain it. An IP address can be bound to only one MAC address. To bind an IP address in a binding entry to another MAC address, you must delete the existing binding entry and then create the new one.

IP-MAC binding tasks at a glance

To configure IP-MAC binding, perform the following tasks:

1. Enabling the IP-MAC binding feature
2. Configuring IP-MAC binding entries

   Choose the options to configure as needed:

   - Manually creating an IP-MAC binding entry
   - Bulk generating IP-MAC binding entries
3. Setting the default action for packets that do not match any IP-MAC binding entries

Enabling the IP-MAC binding feature

About this task

With this feature enabled, the device compares the source IP address and source MAC address in packets with existing IP-MAC binding entries. Packets that do not exactly match any IP-MAC binding entries are dropped.

Procedure

1. Enter system view.

   `system-view`

2. Enable the IP-MAC binding feature.

   `ip-mac binding enable`

   By default, the IP-MAC binding feature is disabled.

Manually creating an IP-MAC binding entry

Creating an IP-MAC binding entry

1. Enter system view.

   `system-view`

2. Create an IP-MAC binding entry.

   IPv4:

   ip-mac binding ipv4 *ipv4-address* **mac-address** *mac-address* [**vlan** *vlan-id* | **vpn-instance** *vpn-instance-name*]

   IPv6:

   ip-mac binding ipv6 *ipv6-address* **mac-address** *mac-address* [**vlan** *vlan-id* | **vpn-instance** *vpn-instance-name*]

   By default, no IP-MAC binding entry is configured.

Bulk generating IP-MAC binding entries

About this task

This task allows the device to generate IP-MAC binding entries in bulk based on existing ARP and ND entries on an interface.

Procedure

1. Enter system view.

   `system-view`

2. Bulk generate IP-MAC binding entries.

   ip-mac binding interface *interface-type interface-number*

Setting the default action for packets that do not match any IP-MAC binding entries

About this task

By default, the device permits packets that do not match any IP-MAC binding entries to pass through. This task allows you to set the default action to deny for these packets.

Procedure

1. Enter system view.
   `system-view`
2. Set the default action to deny for packets that do not match any IP-MAC binding entries.
   `ip-mac binding no-match action deny`
   By default, the action for packets that do not match any IP-MAC binding entries is **permit**.

Display and maintenance commands for IP-MAC binding

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display IPv4-MAC binding entries.	**display ip-mac binding ipv4** [*ipv4-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id* | **vpn-instance** *vpn-instance-name*]
Display IPv6-MAC binding entries.	**display ip-mac binding ipv6** [*ipv6-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id* | **vpn-instance** *vpn-instance-name*]
Display statistics about packets dropped by the IP-MAC binding feature.	**display ip-mac binding statistics** [**slot** *slot-number*]
Display the status of the IP-MAC binding feature.	**display ip-mac binding status**
Clear statistics about packets dropped by the IP-MAC binding feature.	**reset ip-mac binding statistics** [**slot** *slot-number*]

IP-MAC binding configuration examples

Example: Configuring IPv4-MAC binding

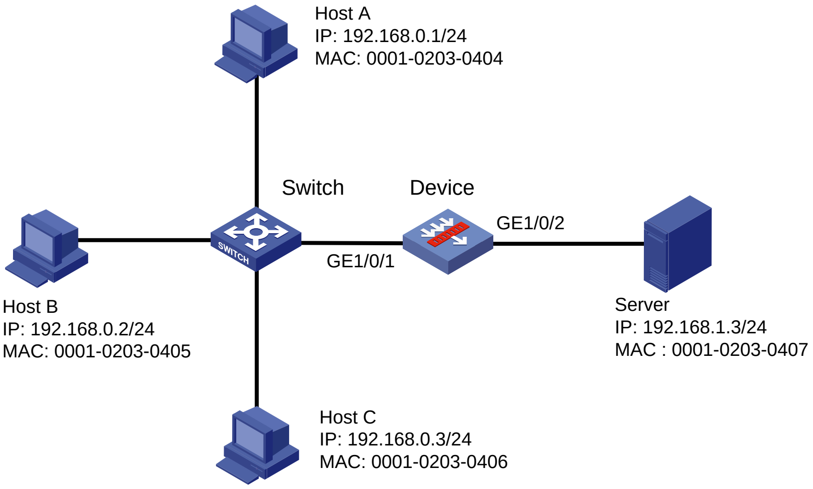
Network configuration

As shown in Figure 2, Host A, Host B, and the server are statically assigned IPv4 addresses. Host A and Host B communicate with the server through the gateway (the device).

Create the following IPv4-MAC binding entries on the device to permit packets only from Host A, Host B, and the server:

- Bind IPv4 address 192.168.0.1 to MAC address 0001-0203-0404 for Host A.
- Bind IPv4 address 192.168.0.2 to MAC address 0001-0203-0405 for Host B.
- Bind IPv4 address 192.168.1.3 to MAC address 0001-0203-0407 for the server.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.0.254 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow hosts in security zone **trust** to access the server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-1-trust-dmz] source-zone trust
[Device-security-policy-ip-1-trust-dmz] destination-zone dmz
[Device-security-policy-ip-1-trust-dmz] source-ip-subnet 192.168.0.0 24
[Device-security-policy-ip-1-trust-dmz] destination-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-dmz] action pass
[Device-security-policy-ip-1-trust-dmz] quit
```

   # Configure a rule named **dmz-trust** to allow the hosts to access the internal servers.

```
[Device-security-policy-ip] rule name untrust-dmz
[Device-security-policy-ip-2-untrust-dmz] source-zone untrust
```

```
[Device-security-policy-ip-2-untrust-dmz] destination-zone dmz
[Device-security-policy-ip-2-untrust-dmz] destination-ip-host 10.1.1.2
[Device-security-policy-ip-2-untrust-dmz] action pass
[Device-security-policy-ip-2-untrust-dmz] quit
[Device-security-policy-ip] quit
```

4. Configure the IP-MAC binding feature:

   # Enable the IP-MAC binding feature on GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

   ```
   [Device] ip-mac binding enable
   ```

   # Create IPv4-MAC binding entries to permit packets only from Host A, Host B, and the server.

   ```
   [Device] ip-mac binding ip 192.168.0.1 mac-address 0001-0203-0404
   [Device] ip-mac binding ip 192.168.0.2 mac-address 0001-0203-0405
   [Device] ip-mac binding ip 192.168.1.3 mac-address 0001-0203-0407
   ```

   # Set the default action to deny for packets that do not match any IP-MAC binding entries.

   ```
   [Device] ip-mac binding no-match action deny
   ```

Verifying the configuration

# Display IPv4-MAC binding entries.

```
<Device> display ip-mac binding ipv4
Total entries: 1
IP address      MAC address         VPN instance         VLAN ID
192.168.0.1     0001-0203-0404      public               N/A
192.168.0.2     0001-0203-0405      public               N/A
192.168.1.3     0001-0203-0407      public               N/A
```

# Ping the server from Host C.

```
C:\> ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.3:
   Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

The ping requests timed out, which indicates that the requests are blocked by the device.

Example: Configuring IPv6-MAC binding

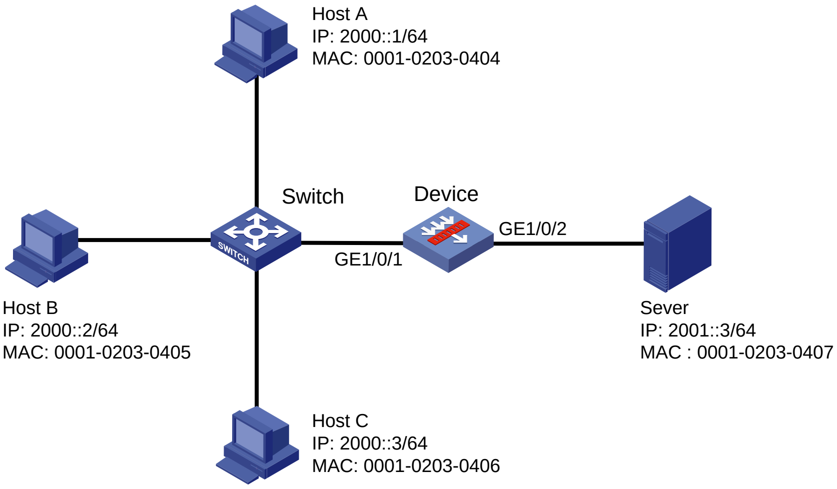
Network configuration

As shown in Figure 3, Host A, Host B, and the server are statically assigned IPv6 addresses. Host A and Host B communicate with the server through the gateway (the device).

Create the following IPv6-MAC binding entries on the device to permit packets only from Host A, Host B, and the server:

- Bind IPv6 address 2000::1/64 to MAC address 0001-0203-0404 for Host A.
- Bind IPv6 address 2000::2/64 to MAC address 0001-0203-0405 for Host B.
- Bind IPv6 address 2001::3/64 to MAC address 0001-0203-0407 for the server.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ipv6 address 2000::4 64
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-dmz** to allow hosts in security zone **trust** to access the server.

```
[Device] security-policy ipv6
[Device-security-policy-ipv6] rule name trust-dmz
[Device-security-policy-ipv6-1-trust-dmz] source-zone trust
[Device-security-policy-ipv6-1-trust-dmz] destination-zone dmz
[Device-security-policy-ipv6-1-trust-dmz] source-ip-subnet 2000::1 64
[Device-security-policy-ipv6-1-trust-dmz] destination-ip-subnet 2001::1 64
[Device-security-policy-ipv6-1-trust-dmz] action pass
[Device-security-policy-ipv6-1-trust-dmz] quit
```

   # Configure a rule named **dmz-trust** to allow the hosts to access the internal servers.

```
[Device-security-policy-ipv6] rule name dmz-trust
[Device-security-policy-ipv6-2-dmz-trust] source-zone dmz
[Device-security-policy-ipv6-2-dmz-trust] destination-zone trust
[Device-security-policy-ipv6-2-dmz-trust] source-ip-subnet 2001::1 64
[Device-security-policy-ipv6-2-dmz-trust] destination-ip-subnet 2000::1 64
[Device-security-policy-ipv6-2-dmz-trust] action pass
```

```
[Device-security-policy-ipv6-2-dmz-trust] quit
[Device-security-policy-ipv6] quit
```

4. Configure the IP-MAC binding feature:

   # Enable the IP-MAC binding feature on GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

   ```
   [Device] ip-mac binding enable
   ```

   # Create IPv6-MAC binding entries to permit packets only from Host A, Host B, and the server.

   ```
   [Device] ip-mac binding ipv6 2000::1 mac-address 0001-0203-0404
   [Device] ip-mac binding ipv6 2000::2 mac-address 0001-0203-0405
   [Device] ip-mac binding ipv6 2001::3 mac-address 0001-0203-0407
   ```

   # Set the default action to deny for packets that do not match any IP-MAC binding entries.

   ```
   [Device] ip-mac binding no-match action deny
   ```

Verifying the configuration

# Display IPv6-MAC binding entries.

```
<Device> display ip-mac binding ipv6
Total entries: 1
IP address      MAC address         VPN instance         VLAN ID
2000::1         0001-0203-0404      public               N/A
2000::2         0001-0203-0405      public               N/A
2001::3         0001-0203-0407      public               N/A
```

# Ping the server from Host C.

```
C:\> ping 2001::3

Pinging 2001::3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 2001::3:
   Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

The ping requests timed out, which indicates that the requests are blocked by the device.

Contents

Configuring APR

About APR

The application recognition (APR) feature recognizes application protocols of packets for features such as QoS, ASPF, and bandwidth management.

APR uses the following methods to recognize an application protocol:

- Port-based application recognition (PBAR).
- Network-based application recognition (NBAR).

PBAR

PBAR maps a port to an application protocol and recognizes packets of the application protocol according to the port-protocol mapping.

PBAR supports the following port-protocol mappings:

- **Predefined**—An application protocol uses the port defined by the system. You can modify the predefined mappings as needed.
- **User-defined**—An application protocol uses the port defined by the user.

PBAR offers the following mappings to maintain and apply user-defined port configuration:

- **General port mapping**—Maps a user-defined port to an application protocol. All packets destined for that port are regarded as packets of the application protocol. For example, if port 2121 is mapped to FTP, all packets destined for that port are regarded as FTP packets.
- **Host-port mapping**—Maps a user-defined port to an application protocol for packets to or from some specific hosts. For example, you can establish a host-port mapping so that all packets destined for the network segment 10.110.0.0/16 on port 2121 are regarded as FTP packets. To define the range of the hosts, you can specify the ACL, the host IP address range, or the subnet.

  Host-port mapping can be further divided into the following categories:

  - **ACL-based host-port mapping**—Maps a port to an application protocol for the packets matching the specified ACL.
  - **Subnet-based host-port mapping**—Maps a port to an application protocol for the packets sent to the specified subnet.
  - **IP address-based host-port mapping**—Maps a port to an application protocol for the packets destined for the specified IP addresses.

APR selects a port mapping to recognize the application protocol of a packet in the following order:

- IP address-based port mapping.
- Subnet-based port mapping.
- ACL-based host-port mapping.
- General port mapping.

For the same type of mappings, the port mapping with a transport layer protocol has higher priority than the mapping without a transport layer protocol.

NBAR

NBAR uses predefined or user-defined NBAR rules to match packet contents to recognize the application protocols of packets that match the applied object policy.

NBAR can recognize the following application types:

- **Predefined**—Defined by NBAR rules in the APR signature library.
- **User-defined**—Defined by user-configured NBAR rules.

Application group

You can add application protocols that have similar signatures or restrictions to an application group. APR recognizes packets of the application protocols by matching the packet contents with the signatures or restrictions. If a packet is recognized as the packet of an application protocol in the application group, the packet is considered to be the packet of the application group. Features such as ASPF and bandwidth management can handle packets belonging to the same group in batch.

You can add application protocols to an application group by using the following methods:

- Add application protocols one by one to the application group.
- Copy application protocols from another application group to the application group.

APR signature library management

APR signature library

APR signature library is a resource library of character string signatures for application recognition. It includes PBAR and NBAR signatures. To meet the changing requirements for application recognition, you must update the APR signature library in a timely manner and roll back the APR signature library as needed.

APR signature library update

You can update the APR signature library by using one of the following methods:

- Automatic update.

  The device automatically downloads the most up-to-date APR signature file to update its local signature library periodically.
- Triggered update.

  The device downloads the most up-to-date APR signature file to update its local signature library immediately after you trigger the update operation.
- Manual update.

  Use this method when the device cannot obtain the APR signature file automatically.

  You must first download the most up-to-date APR signature file manually. The device then obtains the downloaded file to update its local signature library.

APR signature library rollback

You can perform the rollback operation if high error rate or abnormality occurs when the device uses the current APR signature library for application recognition.

You can roll back the current APR signature library to the last version or to the factory version.

Restrictions: Licensing requirements for APR

To update the APR signature library, you must purchase and install the appropriate license. After the license expires, APR can still use the existing signature library but cannot update the signature library. For information about licenses, see license management in *Fundamentals Configuration Guide*.

APR tasks at a glance

To configure APR, perform the following tasks:

1. Configuring PBAR
2. Configuring a user-defined NBAR rule
3. Configuring a risk type for a user-defined application
4. Configuring application groups
5. Enabling application statistics on an interface
6. Configuring detection thresholds for categorizing an application as type other
7. Managing the APR signature library

Configuring PBAR

1. Enter system view.

   system-view

2. Configure a port mapping.

   Choose the options to configure as needed:

   - Configure a general port mapping:

     port-mapping application *application-name* **port** *port-number* [**protocol** *protocol-name*]

   - Configure an ACL-based host-port mapping:

     port-mapping application *application-name* **port** *port-number* [**protocol** *protocol-name*] **acl** [**ipv6**] *acl-number*

   - Configure a subnet-based host-port mapping:

     port-mapping application *application-name* **port** *port-number* [**protocol** *protocol-name*] **subnet** { **ip** *ipv4-address* { *mask-length* | *mask* } | **ipv6** *ipv6-address prefix-length* } [**vpn-instance** *vpn-instance-name*]

   - Configure an IP address-based host-port mapping:

     port-mapping application *application-name* **port** *port-number* [**protocol** *protocol-name*] **host** { **ip** | **ipv6** } *start-ip-address* [*end-ip-address*] [**vpn-instance** *vpn-instance-name*]

   By default, all application protocols are mapped to well-known ports.

   If the specified application protocol does not exist, the system first creates the protocol.

Configuring a user-defined NBAR rule

About this task

You can configure user-defined NBAR rules if predefined NBAR rules cannot meet user needs. The predefined NBAR rules cannot be deleted or modified.

A user-defined NBAR rule can contain the following match criteria:

- Signatures.
- Destination IP subnet.
- Source IP subnet.
- Direction at which the application is recognized.

- Port number.

You can configure more than one match criterion for the NBAR rule. To match the NBAR rule, packets must match all the configured match criteria in the rule. If multiple signatures are configured, packets must match a minimum of one signature.

Procedure

1. Enter system view.

   system-view

2. Create a user-defined NBAR rule and enter its view.

   nbar application *application-name* **protocol** { **http** | **tcp** | **udp** }

3. (Optional.) Configure the description of the NBAR rule.

   description *text*

   By default, the user-defined NBAR rule is described as **User defined application**.

4. Configure a signature and enter NBAR rule signature view.

   signature [*signature-id*] [**field** *field-name*] [**offset** *offset-value*] { **hex** *hex-vector* | **regex** *regex-pattern* | **string** *string* }

   By default, no signatures are configured for an NBAR rule.

5. (Optional.) Configure a detection item for the signature.

   detection *detection-id* **field** *field-name* **match-type** { **exclude** | **include** } { **hex** *hex-vector* | **regex** *regex-pattern* | **text** *text-string* } [**offset** *offset-value* [**depth** *depth-value*] | **relative-offset** *relative-offset-value* [**relative-depth** *relative-depth-value*]]

   By default, no detection items are configured for a signature.

6. Return to user-defined NBAR rule view.

   quit

7. (Optional.) Specify a destination IP subnet.

   destination ip *ipv4-address* [*mask-length*]

   By default, an NBAR rule matches packets with any destination IP address.

8. (Optional.) Specify a source IP subnet.

   source ip *ipv4-address* [*mask-length*]

   By default, an NBAR rule matches packets with any source IP address.

9. (Optional.) Specify a direction.

   direction { **to-client** | **to-server** }

   By default, an NBAR rule matches packets in both directions.

10. (Optional.) Specify a port number or port range.

    service-port { *port-num* | **range** *start-port end-port* }

    By default, an NBAR rule matches packets of all port numbers.

11. (Optional.) Set the maximum detected length.

    apr set detectlen *bytes*

    By default, the maximum detected length is not set for an NBAR rule.

12. (Optional.) Disable the user-defined NBAR rule.

    disable

    By default, a user-defined NBAR rule is enabled.

13. Activate the user-defined NBAR rule.

    inspect activate

For information about this command, see DPI engine commands in *DPI Command Reference*.

Configuring a risk type for a user-defined application

About this task

A user-defined application can have multiple or no risk types.

The more risk types a user-defined application has, the higher risk level the application has. You can configure security policies according to the risk level.

The risk types for predefined applications are automatically generated by the APR signature library.

Restrictions and guidelines

Before configuring risk types, you must update the APR signature library to the latest version.

The user-defined application must already exist.

Procedure

1. Enter system view.

   `system-view`

2. Enter user-defined application view.

   `user-defined-application` *`application-name`*

3. Configure a risk type for the user-defined application.

   `risk type` *`risk-type`*

   By default, a user-defined application does not have any risk type.

Configuring application groups

1. Enter system view.

   `system-view`

2. Create an application group and enter its view.

   `app-group` *`group-name`*

3. (Optional.) Configure the description of the application group.

   `description` *`text`*

   By default, the description is **"User-defined application group"**.

4. Add application protocols to the group.

   Choose the options to configure as needed:

   - Copy all application protocols from another group to the group.

     `copy app-group` *`group-name`*

     Execute this command multiple times to copy application protocols from multiple groups to the current group.

   - Add an application protocol to the group.

     `include application` *`application-name`*

     By default, an application group does not contain any application protocols.

Enabling application statistics on an interface

About this task

When the application statistics feature is enabled on an interface, the device separately counts the number of packets or bytes that the interface has received or sent for each application protocol. It also calculates the transmission rates of the interface for these protocols.

To display application statistics, use the **display application statistics** command.

Restrictions and guidelines

The application statistics feature consumes a large amount of system memory. When the system generates an alarm for lack of memory, disable the application statistics feature on all interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable application statistics on the interface.

   application statistics enable [**inbound** | **outbound**]

   By default, this feature is disabled.

   If you do not specify the **inbound** or **outbound** keyword, this command enables the application statistics feature in both the inbound and outbound directions of the interface.

Configuring detection thresholds for categorizing an application as type other

About this task

If the device cannot identify the application to which the packets of a protocol belongs after detection thresholds are reached, it categorizes the packets as belonging to type **other**.

Restrictions and guidelines

You can configure both the packet count threshold and the payload length threshold for the same protocol.

Procedure

1. Enter system view.

   system-view

2. Configure detection thresholds for categorizing an application as type **other**.

   apr protocol *protocol-name* **detect-threshold** { **packet-count** *count* | **payload-length** *length* } **application-other**

   By default, the device uses predefined detection thresholds in the signature library for categorizing an application as type **other**.

Managing the APR signature library

Restrictions and guidelines for APR signature library management

For a successful APR signature library update or rollback, do not delete the **/dpi/** folder in the root directory on the device storage media.

Do not update or roll back the APR signature library when the remaining system memory reaches any alarm threshold. Insufficient memory causes update or rollback failure and affects the operation of NBAR. For information about memory alarm thresholds, see device management in *Fundamentals Configuration Guide*.

You can update only one APR signature library at a time. If an APR signature library is being updated, please wait for the update to complete before updating another APR signature library.

Scheduling an automatic update for the APR signature library

About this task

If the device can access the signature library services on the official website, you can schedule an automatic update. The automatic update enables the device to automatically update the local APR signature library at the scheduled update time.

Restrictions and guidelines

For a successful automatic update, make sure the following requirements are met:

- The device can obtain the IP address of the official website through static or dynamic domain name resolution.
- The device can access the signature library services on the official website.

For information about DNS, see *Layer 3—IP Services Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable the automatic update feature and enter auto-update configuration view.

   apr signature auto-update

   By default, the automatic update feature is disabled.

3. Configure the update schedule.

   update schedule { **daily** | **weekly** { **fri** | **mon** | **sat** | **sun** | **thu** | **tue** | **wed** } } **start-time** *time* **tingle** *minutes*

   By default, the device automatically updates the APR signature library between 02:01:00 to 04:01:00 every day.

4. (Optional.) Overwrite the current signature file.

   override-current

   By default, the current APR signature file is not overwritten for an update operation. Instead, the device will back up the current APR signature file.

Triggering an automatic update for the APR signature library

About this task

Anytime you find a release of new signature version on the official website, you can trigger the device to immediately update the local APR signature library.

Restrictions and guidelines

For a successful triggered update, make sure the following requirements are met:

- The device can obtain the IP address of the official website through static or dynamic domain name resolution.
- The device can access the signature library services on the official website.

For information about DNS, see *Layer 3—IP Services Configuration Guide.*

Procedure

1. Enter system view.
   `system-view`
2. Trigger an automatic update for the APR signature library.
   `apr signature auto-update-now`

Performing a manual update for the APR signature library

About this task

If the device cannot access the signature library services on the official website, use one of the following methods to manually update the APR signature library on the device:

- **Local update**—By using the locally stored APR signature file.

  The APR signature file must be stored on the mater device for a successful update.
- **FTP/TFTP update**—By using the APR signature file stored on the FTP or TFTP server.

Procedure

1. Enter system view.
   `system-view`
2. Manually update the APR signature library.
   `apr signature update` [**`override-current`**] *`file-path`*

Rolling back the APR signature library

About this task

Each time a rollback operation is performed, the device backs up the APR signature library of the current version. If you repeat the rollback to the last version operation multiple times, the APR signature library will repeatedly switch between the current version and the last version.

Restrictions and guidelines

To ensure that the APR signature library can be successfully rolled back to the last version, back up the current APR signature library each time you update the library.

Procedure

1. Enter system view.
   `system-view`

2. Roll back the APR signature library.
   apr signature rollback { **factory** | **last** }

Display and maintenance commands for APR

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about application groups.	**display app-group** [**name** *group-name*]
Display information about application protocols.	**display application** [**name** *application-name* | **pre-defined** | **user-defined**]
Display statistics for application protocols.	**display application statistics** [**direction** { **inbound** | **outbound** } | **interface** *interface-type interface-number* [**slot** *slot-number*] | **name** *application-name*] *
Display statistics for application protocols on an interface in descending order based on the specified criteria.	**display application statistics top** *number* { **bps** | **bytes** | **packets** | **pps** } **interface** *interface-type interface-number* [**slot** *slot-number*]
Display detection threshold settings for applications categorized as type **other**.	**display apr protocol** [*protocol-name*] **detection-threshold-other**
Display APR signature library information.	**display apr signature library**
Display information about predefined port mappings.	**display port-mapping pre-defined**
Display information about user-defined port mappings.	**display port-mapping user-defined** [**application** *application-name* | **port** *port-number*]
Clear application statistics for interfaces.	**reset application statistics** [**interface** *interface-type interface-number*]

Contents

Configuring keychains

About keychains

A keychain, a sequence of keys, provides dynamic authentication to ensure secure communication by periodically changing the key and authentication algorithm without service interruption.

Operating mechanism

Each key in a keychain has a key string, authentication algorithm, sending lifetime, and receiving lifetime. When the system time is within the lifetime of a key in a keychain, an application uses the key to authenticate incoming and outgoing packets. The keys in the keychain take effect one by one according to the sequence of the configured lifetimes. In this way, the authentication algorithms and keys are dynamically changed to implement dynamic authentication.

Time modes

A keychain operates in absolute time mode or periodic time mode. The lifetime for a key varies by time mode.

- **Absolute time mode**—Each time point in a key's lifetime is in UTC and is not affected by the system's time zone or daylight saving time.
- **Periodic time mode**—A key's lifetime is calculated based on the local time and is affected by the system's time zone and daylight saving time.
  - **daily**—The lifetime for a key is from the specified start time to the specified end time of each day.
  - **weekly**—The lifetime for a key is from the specified start day to the specified end day of each week.
  - **monthly**—The lifetime for a key is from the specified start date to the specified end date of each month.
  - **yearly**—The lifetime for a key is from the specified start month to the specified end month of each year.

Restrictions and guidelines: Keychain configuration

To make sure only one key in a keychain is used at a time to authenticate packets to a peer, set non-overlapping sending lifetimes for the keys in the keychain.

The keys used by the local device and the peer device must have the same authentication algorithm and key string.

Configuring a keychain in absolute time mode

1. Enter system view.

   ```
   system-view
   ```

2. Create a keychain and enter keychain view.

   ```
   keychain keychain-name mode absolute
   ```

3. (Optional.) Configure TCP authentication.
   - Set the kind value in the TCP Enhanced Authentication Option.

     tcp-kind *kind-value*

     By default, the kind value is 254.
   - Set an algorithm ID for a TCP authentication algorithm.

     tcp-algorithm-id { **hmac-md5** | **hmac-sha-256** | **hmac-sm3** | **md5** | **sm3** } *algorithm-id*

     By default, the algorithm ID is 3 for the MD5 authentication algorithm, 5 for the HMAC-MD5 authentication algorithm, 7 for the HMAC-SHA-256 authentication algorithm, 51 for the SM3 authentication algorithm, and 52 for the HMAC-SM3 authentication algorithm.

   When the local device uses TCP to communicate with a peer device from another vendor, make sure both devices have the same kind value and algorithm ID settings. If they do not, modify the settings on the local device.
4. (Optional.) Set a tolerance time for accept keys in the keychain.

   accept-tolerance { *value* | **infinite** }

   By default, no tolerance time is configured for accept keys in a keychain.

   If authentication information is changed, information mismatch occurs on the local and peer devices, and the service might be interrupted. Use this command to ensure continuous packet authentication.
5. Create a key and enter key view.

   key *key-id*
6. Configure the key.
   - Specify an authentication algorithm for the key.

     authentication-algorithm { **hmac-md5** | **hmac-sha-1** | **hmac-sha-256** | **hmac-sm3** | **md5** | **sm3** }

     By default, no authentication algorithm is specified for a key.
   - Configure a key string for the key.

     key-string { **cipher** | **plain** } *string*

     By default, no key string is configured.
   - Set the sending lifetime in UTC mode for the key.

     send-lifetime utc *start-time start-date* { **duration** { *duration-value* | **infinite** } | **to** *end-time end-date* }

     By default, the sending lifetime is not configured for a key.
   - Set the receiving lifetime in UTC mode for the key.

     accept-lifetime utc *start-time start-date* { **duration** { *duration-value* | **infinite** } | **to** *end-time end-date* }

     By default, the receiving lifetime is not configured for a key.
   - (Optional.) Specify the key as the default send key.

     default-send-key

     By default, a keychain does not have a default send key.

     You can specify only one key as the default send key in a keychain.

Configuring a keychain in periodic time mode

1. Enter system view.

   system-view

2. Create a keychain and enter keychain view.

   keychain *keychain-name* **mode periodic** { **daily** | **monthly** | **weekly** | **yearly** }

3. (Optional.) Configure TCP authentication.

   - Set the kind value in the TCP Enhanced Authentication Option.

     tcp-kind *kind-value*

     By default, the kind value is 254.

   - Set an algorithm ID for a TCP authentication algorithm.

     tcp-algorithm-id { **hmac-md5** | **hmac-sha-256** | **hmac-sm3** | **md5** | **sm3** } *algorithm-id*

     By default, the algorithm ID is 3 for the MD5 authentication algorithm, 5 for the HMAC-MD5 authentication algorithm, 7 for the HMAC-SHA-256 authentication algorithm, 51 for the SM3 authentication algorithm, and 52 for the HMAC-SM3 authentication algorithm.

   When the local device uses TCP to communicate with a peer device from another vendor, make sure both devices have the same kind value and algorithm ID settings. If they do not, modify the settings on the local device.

4. (Optional.) Set a tolerance time for accept keys in the keychain.

   accept-tolerance { *value* | **infinite** }

   By default, no tolerance time is configured for accept keys in a keychain.

   If authentication information is changed, information mismatch occurs on the local and peer devices, and the service might be interrupted. Use this command to ensure continuous packet authentication.

5. Create a key and enter key view.

   key *key-id*

6. Configure the key.

   - Specify an authentication algorithm for the key.

     authentication-algorithm { **hmac-md5** | **hmac-sha-1** | **hmac-sha-256** | **hmac-sm3** | **md5** | **sm3** }

     By default, no authentication algorithm is specified for a key.

   - Configure a key string for the key.

     key-string { **cipher** | **plain** } *string*

     By default, no key string is configured.

   - Set the daily, weekly, monthly, or yearly sending lifetime in periodic time mode for the key.

     send-lifetime daily *start-day-time* **to** *end-day-time*

     send-lifetime date { *month-day*&<1-31> | *start-month-day* **to** *end-month-day* }

     send-lifetime day { *week-day* | *start-week-day* **to** *end-week-day* }

     send-lifetime month { *month* | *start-month* **to** *end-month* }

     By default, the sending lifetime is not configured for a key.

   - Set the daily, weekly, monthly, or yearly receiving lifetime in periodic time mode for the key.

     accept-lifetime daily *start-day-time* **to** *end-day-time*

     accept-lifetime date { *month-day*&<1-31> | *start-month-day* **to** *end-month-day* }

     accept-lifetime day { *week-day* | *start-week-day* **to** *end-week-day* }

     accept-lifetime month { *month* | *start-month* **to** *end-month* }

     By default, the receiving lifetime is not configured for a key.

- (Optional.) Specify the key as the default send key.

  default-send-key

  By default, a keychain does not have a default send key.

  You can specify only one key as the default send key in a keychain.

Display and maintenance commands for keychain

Execute **display** commands in any view.

Task	Command
Display keychain information.	**display keychain** [**name** *keychain-name* [**key** *key-id*]]

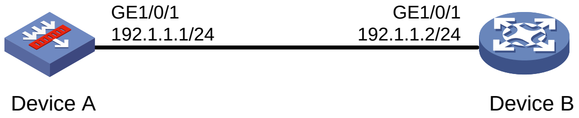
Keychain configuration examples

Example: Configuring keychains

Network configuration

As shown in Figure 1, establish an OSPF neighbor relationship between Device A and Device B, and use a keychain to authenticate packets between the devices. Configure key 1 and key 2 for the keychain and make sure key 2 is used immediately when key 1 expires.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces and configure routes, security zones, zone pairs, and interzone policies. Make sure the network connections are available. (Details not shown.)
2. Configure Device A:

   # Configure OSPF.

   ```
   <DeviceA> system-view
   [DeviceA] ospf 1 router-id 1.1.1.1
   [DeviceA-ospf-1] area 0
   [DeviceA-ospf-1-area-0.0.0.0] network 192.1.1.0 0.0.0.255
   [DeviceA-ospf-1-area-0.0.0.0] quit
   [DeviceA-ospf-1] quit
   ```

   # Create a keychain named **abc**, and specify the absolute time mode for it.

   ```
   [DeviceA] keychain abc mode absolute
   ```

   # Create key **1** for keychain **abc**, specify an authentication algorithm, and configure a key string and the sending and receiving lifetimes for the key.

   ```
   [DeviceA-keychain-abc] key 1
   [DeviceA-keychain-abc-key-1] authentication-algorithm md5
   [DeviceA-keychain-abc-key-1] key-string plain 123456
   ```

```
[DeviceA-keychain-abc-key-1] send-lifetime utc 10:00:00 2015/02/06 to 11:00:00
2015/02/06
[DeviceA-keychain-abc-key-1] accept-lifetime utc 10:00:00 2015/02/06 to 11:00:00
2015/02/06
[DeviceA-keychain-abc-key-1] quit
```

# Create key **2** for keychain **abc**, specify an authentication algorithm, and configure a key string and the sending and receiving lifetimes for the key.

```
[DeviceA-keychain-abc] key 2
[DeviceA-keychain-abc-key-2] authentication-algorithm hmac-md5
[DeviceA-keychain-abc-key-2] key-string plain pwd123
[DeviceA-keychain-abc-key-2] send-lifetime utc 11:00:00 2015/02/06 to 12:00:00
2015/02/06
[DeviceA-keychain-abc-key-2] accept-lifetime utc 11:00:00 2015/02/06 to 12:00:00
2015/02/06
[DeviceA-keychain-abc-key-2] quit
[DeviceA-keychain-abc] quit
```

# Configure GigabitEthernet 1/0/1 to use keychain **abc** for authentication.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ospf authentication-mode keychain abc
[DeviceA-GigabitEthernet1/0/1] quit
```

3. Configure Device B:

# Configure OSPF.

```
<DeviceB> system-view
[DeviceB] ospf 1 router-id 2.2.2.2
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 192.1.1.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] quit
```

# Create a keychain named **abc**, and specify the absolute time mode for it.

```
[DeviceB] keychain abc mode absolute
```

# Create key **1** for keychain **abc**, specify an authentication algorithm, and configure a key string and the sending and receiving lifetimes for the key.

```
[DeviceB-keychain-abc] key 1
[DeviceB-keychain-abc-key-1] authentication-algorithm md5
[DeviceB-keychain-abc-key-1] key-string plain 123456
[DeviceB-keychain-abc-key-1] send-lifetime utc 10:00:00 2015/02/06 to 11:00:00
2015/02/06
[DeviceB-keychain-abc-key-1] accept-lifetime utc 10:00:00 2015/02/06 to 11:10:00
2015/02/06
[DeviceB-keychain-abc-key-1] quit
```

# Create key **2** for keychain **abc**, specify an authentication algorithm, and configure a key string and the sending and receiving lifetimes for the key.

```
[DeviceB-keychain-abc] key 2
[DeviceB-keychain-abc-key-2] key-string plain pwd123
[DeviceB-keychain-abc-key-2] authentication-algorithm hmac-md5
[DeviceB-keychain-abc-key-2] send-lifetime utc 11:00:00 2015/02/06 to 12:00:00
2015/02/06
[DeviceB-keychain-abc-key-2] accept-lifetime utc 11:00:00 2015/02/06 to 12:00:00
2015/02/06
```

```
[DeviceB-keychain-abc-key-2] quit
[DeviceB-keychain-abc] quit
```

# Configure GigabitEthernet 1/0/1 to use keychain **abc** for authentication.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ospf authentication-mode keychain abc
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

1. When the system time is within the lifetime from 10:00:00 to 11:00:00 on the day 2015/02/06, verify the status of the keys in keychain **abc**.

   # Display keychain information on Device A. The output shows that key 1 is the valid key.

```
[DeviceA] display keychain

 Keychain name          : abc
   Mode                 : absolute
   Accept tolerance     : 0
   TCP kind value       : 254
   TCP algorithm value
     HMAC-MD5           : 5
     HMAC-SHA-256       : 7
     MD5                : 3
     HMAC-SM3           : 52
     SM3                : 51
   Default send key ID  : None
   Active send key ID   : 1
   Active accept key IDs: 1

   Key ID               : 1
     Key string         : $c$3$dYTC8QeOKJkwFwP2k/rWL+1p6uMTw3MqNg==
     Algorithm          : md5
     Send lifetime      : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Send status        : Active
     Accept lifetime    : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Accept status      : Active

   Key ID               : 2
     Key string         : $c$3$7TSPbUxoP1ytOqkdcJ3K3x0BnXEWl4mOEw==
     Algorithm          : hmac-md5
     Send lifetime      : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Send status        : Inactive
     Accept lifetime    : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Accept status      : Inactive
```

   # Display keychain information on Device B. The output shows that key 1 is the valid key.

```
[DeviceB]display keychain

 Keychain name          : abc
   Mode                 : absolute
   Accept tolerance     : 0
   TCP kind value       : 254
   TCP algorithm value
     HMAC-MD5           : 5
     HMAC-SHA-256       : 7
     MD5                : 3
     HMAC-SM3           : 52
     SM3                : 51
   Default send key ID  : None
   Active send key ID   : 1
   Active accept key IDs: 1

   Key ID               : 1
     Key string         : $c$3$/G/Shnh6heXWprlSQy/XDmftHa2JZJBSgg==
     Algorithm          : md5
     Send lifetime      : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Send status        : Active
     Accept lifetime    : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Accept status      : Active

   Key ID               : 2
     Key string         : $c$3$t4qHAw1hpZYN0JKIEpXPcMFMVT81u0hiOw==
     Algorithm          : hmac-md5
     Send lifetime      : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Send status        : Inactive
     Accept lifetime    : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Accept status      : Inactive
```

2. When the system time is within the lifetime from 11:00:00 to 12:00:00 on the day 2015/02/06, verify the status of the keys in keychain **abc**.

   # Display keychain information on Device A. The output shows that key 2 becomes the valid key.

```
[DeviceA]display keychain

 Keychain name          : abc
   Mode                 : absolute
   Accept tolerance     : 0
   TCP kind value       : 254
   TCP algorithm value
     HMAC-MD5           : 5
     HMAC-SHA-256       : 7
     MD5                : 3
     HMAC-SM3           : 52
     SM3                : 51
   Default send key ID  : None
   Active send key ID   : 2
   Active accept key IDs: 2

   Key ID               : 1
     Key string         : $c$3$dYTC8QeOKJkwFwP2k/rWL+1p6uMTw3MqNg==
     Algorithm          : md5
     Send lifetime      : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Send status        : Inactive
     Accept lifetime    : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Accept status      : Inactive

   Key ID               : 2
     Key string         : $c$3$7TSPbUxoP1ytOqkdcJ3K3x0BnXEWl4mOEw==
     Algorithm          : hmac-md5
     Send lifetime      : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Send status        : Active
     Accept lifetime    : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Accept status      : Active
```

# Display keychain information on Device B. The output shows that key 2 becomes the valid key.

```
[DeviceB]display keychain

Keychain name          : abc
   Mode                 : absolute
   Accept tolerance     : 0
   TCP kind value       : 254
   TCP algorithm value
     HMAC-MD5           : 5
     HMAC-SHA-256       : 7
     MD5                : 3
     HMAC-SM3           : 52
     SM3                : 51
   Default send key ID  : None
   Active send key ID   : 1
   Active accept key IDs: 1

   Key ID               : 1
     Key string         : $c$3$/G/Shnh6heXWprlSQy/XDmftHa2JZJBSgg==
     Algorithm          : md5
     Send lifetime      : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Send status        : Inactive
     Accept lifetime    : 10:00:00 2015/02/06 to 11:00:00 2015/02/06
     Accept status      : Inactive

   Key ID               : 2
     Key string         : $c$3$t4qHAw1hpZYN0JKIEpXPcMFMVT81u0hiOw==
     Algorithm          : hmac-md5
     Send lifetime      : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Send status        : Active
     Accept lifetime    : 11:00:00 2015/02/06 to 12:00:00 2015/02/06
     Accept status      : Active
```

Contents

Configuring crypto engines

About crypto engines

Crypto engines encrypt and decrypt data for service modules.

Crypto engine types

Crypto engines include the following types:

- **Hardware crypto engines**—A hardware crypto engine is a coprocessor integrated on a CPU or hardware crypto card. Hardware crypto engines can accelerate encryption/decryption speed, which improves device processing efficiency. You can enable or disable hardware crypto engines globally as needed. By default, hardware crypto engines are enabled.
- **Software crypto engines**—A software crypto engine is a set of software encryption algorithms. The device uses software crypto engines to encrypt and decrypt data for service modules. They are always enabled. You cannot enable or disable software crypto engines.

Crypto engine processing mechanism

If you disable hardware crypto engines, the device uses only software crypto engines for data encryption/decryption. If you enable hardware crypto engines, the device preferentially uses hardware crypto engines. If the device does not support hardware crypto engines, or if the hardware crypto engines do not support the required encryption algorithm, the device uses software crypto engines for data encryption/decryption.

Crypto engines provide encryption/decryption services for service modules, for example, the IPsec module. When a service module requires data encryption/decryption, it sends the desired data to a crypto engine. After the crypto engine completes data encryption/decryption, it sends the data back to the service module.

Display and maintenance commands for crypto engines

Execute **display** commands in any view and **reset** commands in user view.

Task	**Command**
Display crypto engine information.	**display crypto-engine**
Display crypto engine statistics.	**display crypto-engine statistics** [**engine-id** *engine-id* **slot** *slot-number*]
Clear crypto engine statistics.	**reset crypto-engine statistics** [**engine-id** *engine-id* **slot** *slot-number*]

Contents

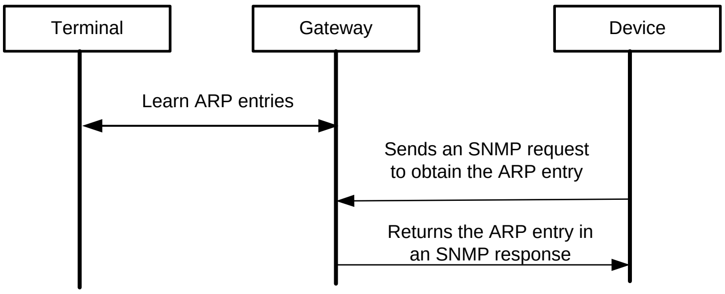
Configuring MAC address learning through a Layer 3 device

About MAC address learning through a Layer 3 device

This feature enables the device to learn the MAC address of a terminal (a PC for example) when a Layer 3 device exists between the device and the terminal through SNMP for network traffic control.

Working mechanism

Figure 1 MAC address learning through a Layer device workflow


Terminal
Gateway
Device
Learn ARP entries
Sends an SNMP request to obtain the ARP entry
Returns the ARP entry in an SNMP response

As shown in Figure 1, MAC address learning through a Layer 3 device proceeds as follows:

1. The gateway learns the IP-MAC binding of the terminal, and then generates an ARP entry.
2. The device sends SNMP requests to the gateway at the specified intervals to request the ARP entry.
3. The gateway sends a response that contains the ARP entry to the device.
4. Upon receiving the response, the device saves the ARP entry in the memory. Then it can learn the MAC address of the terminal.

Entry aging

ARP entries learned through a Layer 3 device will be automatically deleted when the aging timer expires, or can be cleared using the **`reset snmp-server arp-sync table`** command.

Restrictions and guidelines: MAC address learning through a Layer 3 device configuration

- MAC addresses learned using this feature can be used for policy packet filtering and packet information recording in the IPS logs.
- Only MAC addresses mapped from IPv4 addresses can be learned.
- Make sure no NAT devices exist between the device and the Layer 3 device.
- This feature is not applicable to a VRF network.

Prerequistes

Make sure SNMP agent has been enabled and a community name has been configured on the Layer 3 device. For information about SNMP, see SNMP configuration in *Network Management and Monitoring*.

MAC address learning through a Layer 3 device configuration tasks at a glance

To configure MAC address learning through a Layer 3 device, perform the following tasks:

1. Enabling ARP entry synchronization through SNMP
2. Configuring the target Layer 3 device
3. (Optional.) Setting the parameters for ARP entry synchronization

Enabling ARP entry synchronization through SNMP

About this task

With this feature enabled, the device acts as an NMS to learn all ARP entries on a Layer 3 device (agent) to obtain the MAC address of the Layer 3 device.

Procedure

1. Enter system view.

   system-view

2. Enable ARP entry synchronization through SNMP.

   snmp-server arp-sync enable

   By default, ARP entry synchronization through SNMP is disabled.

Configuring the target Layer 3 device

1. Enter system view.

   system-view

2. Configure the target Layer 3 device for ARP entry synchronization through SNMP.

   SNMP v2c:

   snmp-server arp-sync target-host address *ip-address* **community** { **simple** | **cipher** } *community-name* **v2c**

   SNMP v3:

   snmp-server arp-sync target-host address *ip-address* **usm-user v3** *user-name* [{ **simple** | **cipher** } **authentication-mode** { **md5** | **sha** } *auth-password* [**privacy-mode** { **aes128** | **des56** } *pri-password*]]

   By default, no target Layer 3 device is configured for ARP entry synchronization through SNMP.

Setting the parameters for ARP entry synchronization

About this task

With this feature configured, the device sends SNMP requests for ARP entry synchronization to the target Layer 3 device at the specified intervals. If the device does not receive an SNMP response before the timeout expires within the specified interval, the device re-sends SNMP requests.

Procedure

1. Enter system view.

   `system-view`

2. Set the parameters for ARP entry synchronization.

   `snmp-server arp-sync` { **`interval`** *`interval`* | **`timeout`** *`time`* } *

   By default, the interval for sending SNMP requests is 5 seconds and the timeout for SNMP responses is 3 seconds.

Display and maintenance commands for

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display the ARP entries synchronized through SNMP.	**`display snmp-server arp-sync table`**
Clear the ARP entries synchronized through SNMP.	**`reset snmp-server arp-sync table`**

MAC address learning through a Layer 3 device configuration examples

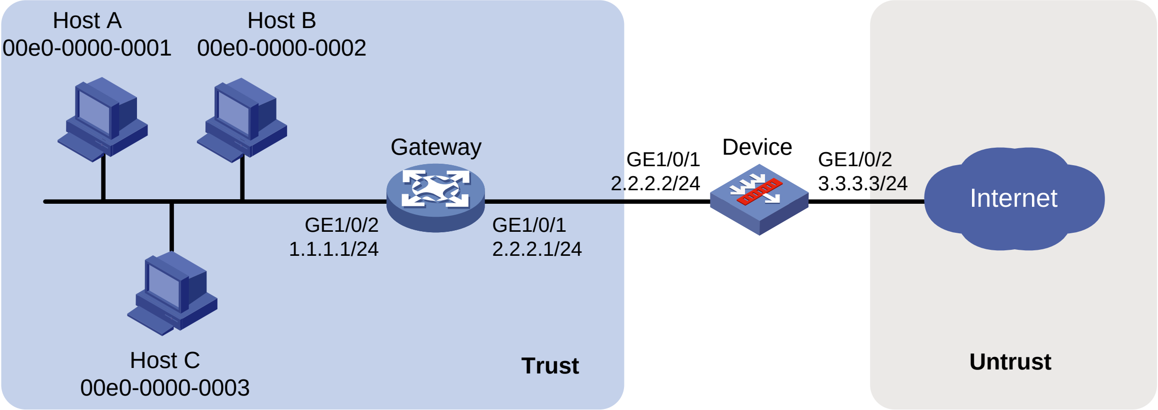
Example: Configuring MAC address learning through a Layer 3 device

Network configuration

As shown in Figure 2, hosts in an internal network are connected to the device through a Layer 3 gateway and the device is connected to the Internet.

- Configure MAC address learning through a Layer 3 device to ensure that the device can learn the MAC addresses of the hosts through the gateway.
- Configure security policies by using the learned MAC addresses to allow only Host A and Host B to access the Internet.

Figure 2 Network diagram



Procedure

1. Configure the gateway:

   # Specify an IP address for each interface and configure routing features to ensure network reachability. (Details not shown.)

   # Configure SNMPv2c, and create the read-only community with the plaintext form name **public**.

```
<Gateway> system-view
[Gateway] snmp-agent sys-info version v2c
[Gateway] snmp-agent community read simple public
```

2. Configure the device:

   a. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Configure settings for routing.

   This example configures static routes, and the next hop in the route to the Internet is 3.3.3.1.

```
[Device] ip route-static 1.1.1.0 24 2.2.2.1
[Device] ip route-static 0.0.0.0 0 3.3.3.1
```

   c. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

   d. Configure a security policy.

   # Configure a rule named **rule1** to allow the device to access the gateway.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name rule1
[Device-security-policy-ip-0-rule1] source-zone local
[Device-security-policy-ip-0-rule1] destination-zone trust
```

```
[Device-security-policy-ip-0-rule1] source-ip-host 2.2.2.2
[Device-security-policy-ip-0-rule1] destination-ip-host 2.2.2.1
[Device-security-policy-ip-0-rule1] action pass
[Device-security-policy-ip-0-rule1] quit
```

# Configure a rule named **rule2** to permit traffic only from MAC object group **groupmac** to the Internet.

```
[Device-security-policy-ip] rule name rule2
[Device-security-policy-ip-1-rule2] source-zone trust
[Device-security-policy-ip-1-rule2] destination-zone untrust
[Device-security-policy-ip-1-rule2] source-mac groupmac
[Device-security-policy-ip-1-rule2] action pass
[Device-security-policy-ip-1-rule2] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

e. Configure MAC address learning through the gateway.

Enable ARP entry synchronization through SNMP. Configure the device to synchronize ARP entries from the gateway.

```
[Device] snmp-server arp-sync enable
[Device] snmp-server arp-sync target-host address 2.2.2.1 community simple public
v2c
[Device] snmp-server arp-sync interval 10 timeout 4
```

f. Create a MAC object group named **groupmac**. Add Host A's MAC address **00e0-0000-0001** and Host B's MAC address **00e0-0000-0002** to the group.

```
[Device] object-group mac-address groupmac
[Device-obj-grp-mac-groupmac] mac 00e0-0000-0001
[Device-obj-grp-mac-groupmac] mac 00e0-0000-0002
[Device-obj-grp-mac-groupmac] quit
```

Verifying the configuration

1. Verify that ARP entries have been synchronized to the device.

```
[Device] display snmp-server arp-sync table
IP Address       MAC Address        Aging(M)
1.1.1.1          00e0-0000-0001     1
1.1.1.2          00e0-0000-0002     1
1.1.1.3          00e0-0000-0003     1
Total:3
```

2. Verify that Host A and Host B can access the external network but Host C cannot.

Contents

SMS

About SMS

Short message service (SMS) is a text messaging service implemented through an SMS gateway. When operating as an SMS gateway, the device uses HTTP to send SMS data to the associated third-party SMS platform. Then, the third-party SMS platform sends SMS messages to users.

Restrictions and guidelines: SMS configuration

The device supports only the Emay SMS platform in the current software version.

Procedure

1. Enter system view.

   system-view

2. Create an SMS gateway and enter its view.

   sms-gateway *gateway-name*

3. Specify the SMS platform for the SMS gateway.

   sms-platform emay

   By default, no SMS platform is specified for an SMS gateway.

4. Configure the app ID for the third-party SMS platform.

   app-id *app-id*

   By default, no app ID is configured for the third-party SMS platform.

5. Configure the secret key for SMS data encryption.

   secret-key { **cipher** | **simple** } *string*

   By default, no secret key is configured for SMS data encryption.

6. (Optional.) Send an SMS message to the test mobile number.

   sms-send test-mobile *number*

7. (Optional.) Specify the VPN instance for the SMS gateway.

   vpn-instance *vpn-instance-name*

   By default, an SMS gateway belongs to the public network.

Display and maintenance commands for SMS

Execute the **display** command in any view.

Task	Command
Display SMS gateway information.	**display sms-gateway** [**brief** | **name** *gateway-name*]

NSFOCUS Firewall Series

NF DPI Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for DPI features, including DPI overview, DPI engine, IPS, URL filtering, data filtering, file filtering, anti-virus, data analysis center, proxy policy.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

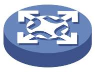
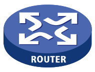
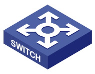
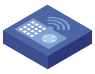
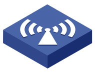
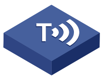
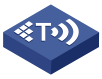
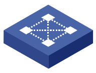
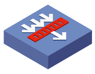
GUI conventions

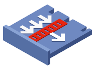
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
(!) **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

DPI overview

About DPI

Deep packet inspection (DPI) inspects application layer payloads to protect the network against application layer malicious activities, such as worms, viruses, spams, breaches, and information leakage.

Traditional security technology relies on the network layer and transport layer. DPI further enhances network security.

DPI functions

DPI provides the following functions:

- **Service identification**—The DPI engine identifies the service of a data flow by analyzing the application layer payload and matching the payload against signatures. DPI engine informs the DPI service modules of the identification results for service control.
- **Service control**—DPI service modules control services flexibly by using DPI service policies. Actions that DPI service policies use for data flows include permit, drop, block source, reset, capture, and log.
- **Service statistics**—DPI provides service statistics about service types, protocol parsing, signature inspection, and packet processing. Service statistics visually display the distribution of data flows and the use of different services. You can find factors that might promote service development or affect network operation.

DPI signature libraries

A DPI signature library is a collection of common signatures that DPI uses for service identification. NSFOCUS releases up-to-date signatures in the form of DPI signature library files. You can manually download the files or configure the device to automatically download the files to update the DPI signature libraries. You can also define signatures of your own as required.

The device supports the following DPI signature libraries:

- IPS signature library.
- URL filtering signature library.
- APR signature library.
- Virus signature library.

DPI services

Table 1 lists the supported DPI services.

Table 1 DPI services

DPI service	Function
IPS	Monitors network traffic for malicious activities and proactively takes actions to protect the network against attacks.
URL filtering	Controls access to the Web resources by filtering the URLs that the users visit.

DPI service	Function
Data filtering	Inspects the content in application protocol packets and filters out illegal packets. With content filtering, you can prevent internal users from accessing inappropriate websites or receiving packets that carry illegal content from the Internet.
File filtering	Filters files by filename extensions.
Anti-virus	Inspects and handles viruses in files to protect the internal network.
NBAR	Identifies the application layer protocols of packets by comparing packet content against signatures. For more information about NBAR, see *Security Configuration Guide*.

DPI mechanism

DPI can be implemented based on security policies.

Security policy-based DPI mechanism

Figure 1 illustrates how security policy-based DPI works.

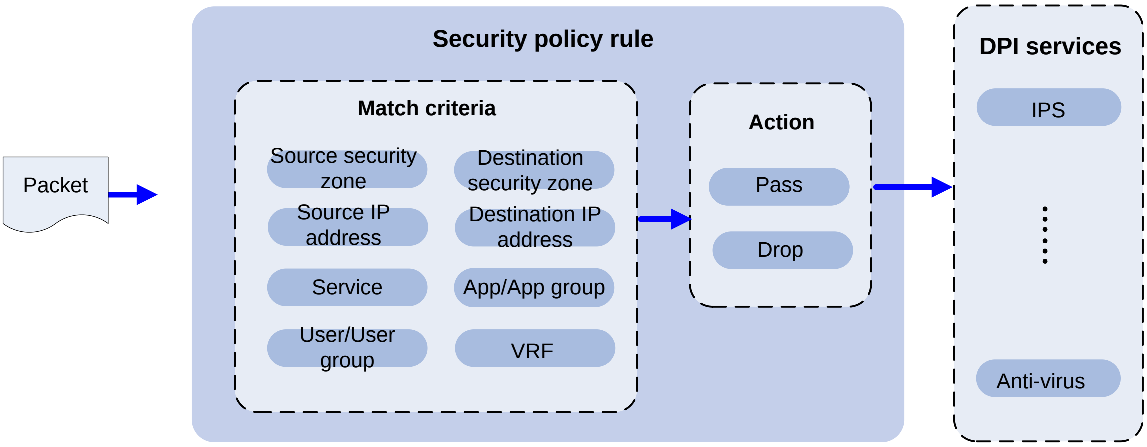
After receiving a packet, the device matches the packet against the configured security policy rules.

A security policy rule includes various match criterion types. A packet matches a policy rule if the packet matches all the criterion types in the rule. Each criterion type includes one or more criteria, and a packet matches a criterion type if it matches any criterion of the type.

For information about security policy rules, see security policy configuration in *Security Configuration Guide*.

- If no matching rule is found, the device drops the packet.
- If a matching rule is found, the device processes the packet according to the rule action:
  - If the rule action is **drop**, the device drops the packet.
  - If the rule action is **pass** and a DPI application profile is specified for the rule, the device uses the DPI application profile to perform DPI on the packet. If the DPI application profile does not exist, the device permits the packet to pass.
  - If the rule action is **pass** and no DPI application profile is specified for the rule, the device permits the packet to pass.

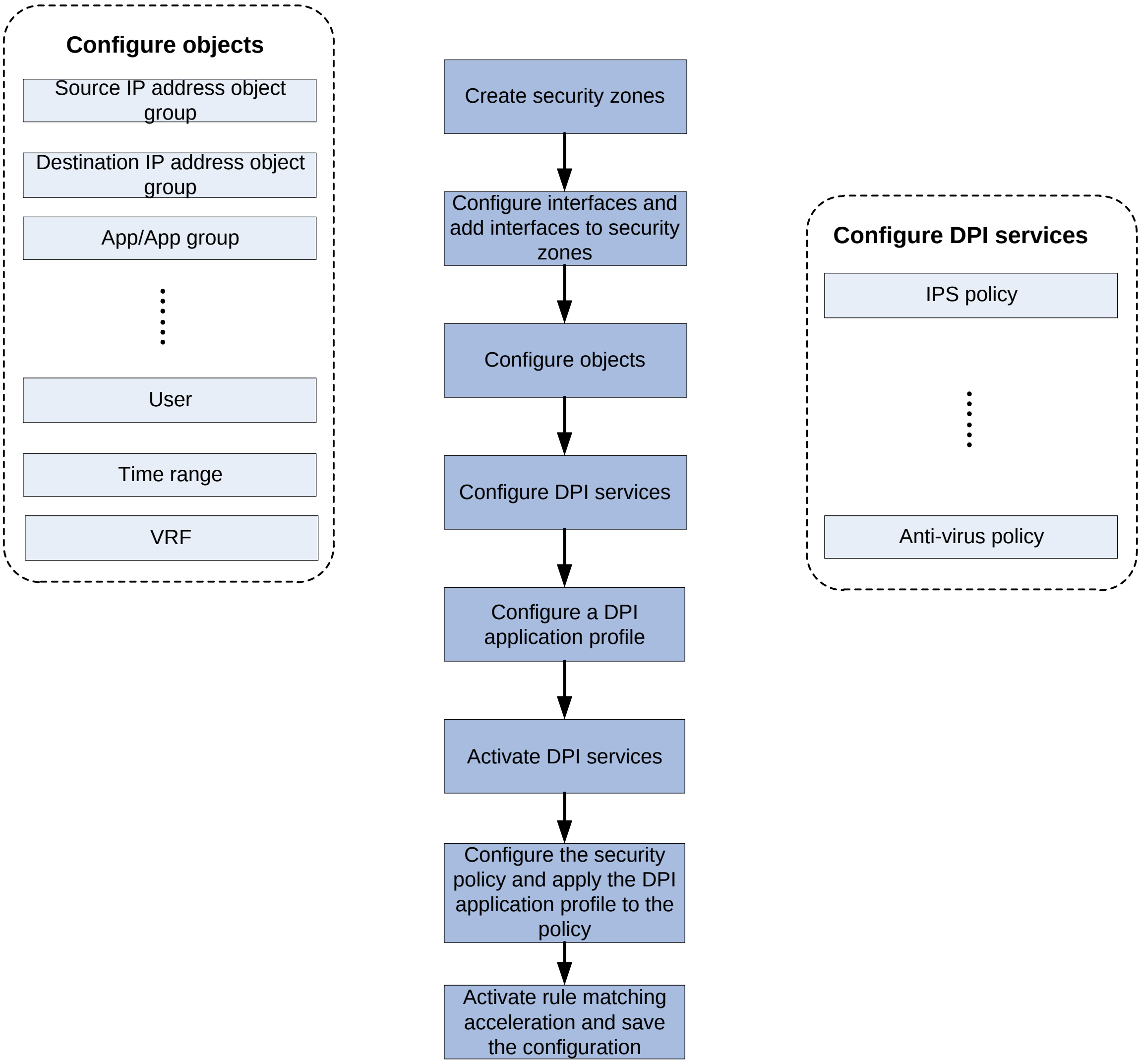
Figure 1 Security policy-based DPI mechanism



DPI configuration workflow

The basic DPI configuration workflow is shown in Figure 2.

Figure 2 DPI configuration workflow


Configure objects
Source IP address object group
Destination IP address object group
App/App group
User
Time range
VRF
Create security zones
Configure interfaces and add interfaces to security zones
Configure objects
Configure DPI services
Configure a DPI application profile
Activate DPI services
Configure the security policy and apply the DPI application profile to the policy
Activate rule matching acceleration and save the configuration
Configure DPI services
IPS policy
Anti-virus policy

Contents

Configuring DPI engine

About DPI engine

DPI engine is an inspection module shared by DPI service modules. DPI engine uses inspection rules to identify the application layer information, including the application layer protocol and behavior. DPI service modules process packets based on the inspection results.

DPI functions

DPI engine provides the following functions:

- **Protocol parsing**—Identifies the application layer protocols and analyzes the application layer information. Information analysis includes recognizing, normalizing, and uncompressing application layer fields.
- **AC pattern matching**—Matches packet payloads by the Aho-Corasick (AC) patterns in inspection rules. AC pattern matching is fast and it is the core function of the DPI engine.
- **Option matching**—Matches packet payloads by the options in the inspection rules whose AC patterns have been matched. Option matching is slower than AC pattern matching.

DPI engine inspection rules

DPI engine uses inspection rules to match packets. Inspection rules are transformed from the rules or signatures of the DPI service modules. The match criteria in an inspection rule can contain the following types:

- **AC pattern**—Criteria that identify packet signatures. An AC pattern is a character string that is three or more bytes long.
- **Option**—Criteria other than AC patterns. For example, an option can be the port number or protocol type.

An inspection rule can contain both AC patterns and options. A packet must match both the AC patterns and options to match the rule.

An inspection rule can also contain only options. A packet matches the rule if it matches the options in the rule.

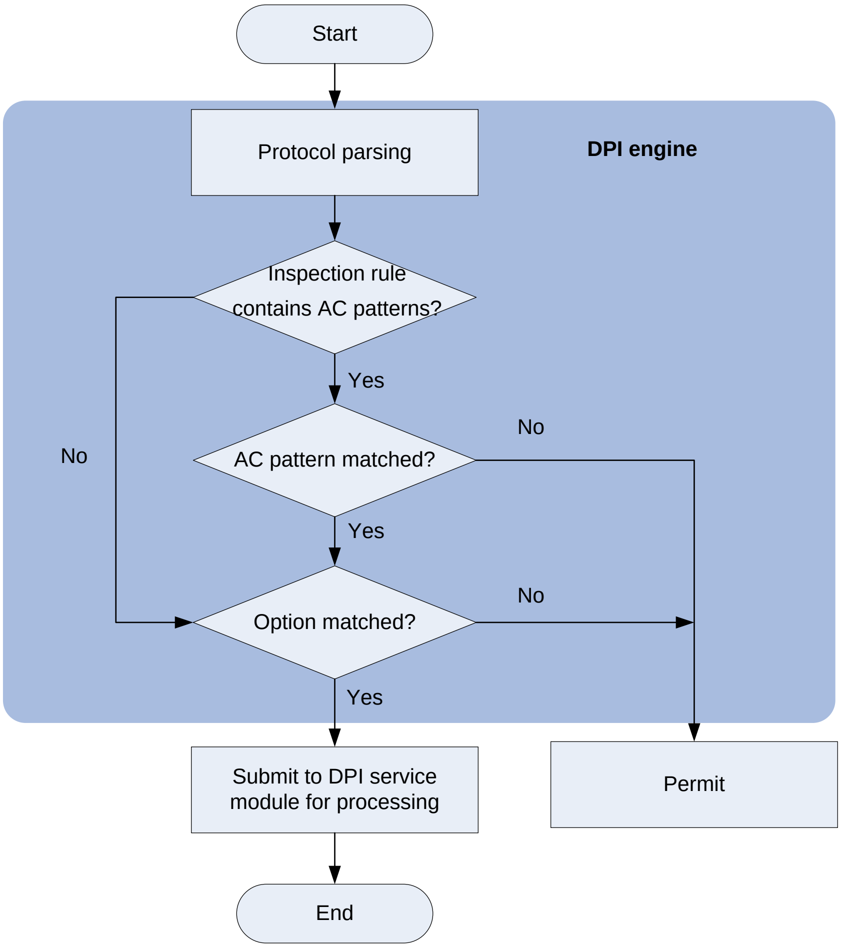
DPI engine mechanism

As shown in Figure 1, DPI engine works as follows:

1. The DPI engine performs protocol parsing for the packet and searches for applicable inspection rules according to the parsing results.
2. If an applicable inspection rule contains AC patterns, DPI engine performs AC pattern matching first. If an applicable inspection rule does not contain AC patterns, DPI engine directly performs option matching. The packet matches the rule if it matches the options.
3. If the packet matches an AC pattern in an applicable inspection rule, the DPI engine further compares the packet against the options associated with the AC pattern. The packet matches the rule if it matches the both the AC pattern and its associated options. If the packet matches an AC pattern but does not match its associated options, the DPI engine permits the packet to pass.

4. If the packet matches an inspection rule, the DPI engine submits the packet to the corresponding DPI service module for processing. If the packet does not match any rule, the DPI engine permits the packet to pass.

Figure 1 DPI engine mechanism



DPI engine tasks at a glance

To configure the DPI engine, perform the following tasks:

1. Configure a DPI application profile
2. Activating policy and rule settings for DPI service modules
3. Configuring action parameter profiles
4. (Optional.) Optimizing the DPI engine
5. (Optional.) Enabling inspection suspension upon excessive CPU usage
6. (Optional.) Configuring DPI engine inspection parameters
   - Configuring an inspection mode
   - Configuring stream fixed length inspection
   - Configuring file fixed length inspection
   - Configuring MD5 fixed-length file inspection
   - Configuring MD5 hash-based virus inspection for all files
   - Setting the maximum data size for file decompression

- Setting the maximum number of file decompression operations
- Setting the maximum number of NFS file names recorded

7. (Optional.) Configuring advanced features of the DPI engine
   - Enabling source port-based application identification
   - Specifying a proxy server for online DPI service signature update
   - Specifying the cloud query server for DPI services
   - Enabling support for HA dual-active mode
   - Configuring real source IP inspection
8. Disabling the DPI engine

Configure a DPI application profile

About this task

A DPI application profile includes a set of DPI service policies, such as a URL filtering policy. It can be applied to a security policy rule to specify the DPI service policy for packets that match the rule.

Procedure

1. Enter system view.

   system-view

2. Create a DPI application profile and enter its view.

   app-profile *profile-name*

3. Apply DPI service policies to the DPI application profile.
   - Specify an IPS policy.

     ips apply policy *policy-name* **mode** { **protect** | **alert** }

     For more information about this command, see IPS commands in *DPI Command Reference*.
   - Specify a URL filtering policy.

     url-filter apply policy *policy-name*

     For more information about this command, see URL filtering commands in *DPI Command Reference*.
   - Specify a data filtering policy.

     data-filter apply policy *policyname*

     For more information about this command, see data filtering commands in *DPI Command Reference*.
   - Specify a file filtering policy.

     file-filter apply policy *policyname*

     For more information about this command, see file filtering commands in *DPI Command Reference*.
   - Specify an anti-virus policy.

     anti-virus apply policy *policyname* **mode** { **alert** | **protect** }

     For more information about this command, see anti-virus commands in *DPI Command Reference*.

   By default, no DPI service policies are applied to a DPI application profile.

Activating policy and rule settings for DPI service modules

About this task

By default, the system will detect whether another configuration change (such as creation, modification, or deletion) occurs within a 20-second interval after a configuration change for DPI service modules such as URL filtering:

- If no configuration change occurs within the interval, the system performs an activation operation at the end of the next interval to make the configuration take effect.
- If a configuration change occurs within the interval, the system continues to periodically check whether a configuration change occurs within the interval.

To activate the policy and rule configurations for DPI service modules immediately, you can execute the **`inspect activate`** command.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Activate policy and rule settings for DPI service modules.

   ```
   inspect activate
   ```

   By default, the creation, modification, and deletion of DPI service policies and rules will be activated automatically.

△ **CAUTION:**

This command causes transient DPI service interruption. DPI-based services might also be interrupted. For example, security policies cannot control access to applications.

Configuring action parameter profiles

Configuring a block source parameter profile

About this task

A block source parameter profile defines the block period for the block source action in DPI service modules.

Restrictions and guidelines

The block source action takes effect only after the blacklist feature is enabled.

With the blacklist feature is enabled, the device drops the matching packet and adds the packet's source IP address to the IP blacklist. Subsequent packets from the source IP address will be dropped directly during the block period.

For more information about the blacklist feature, see attack detection and prevention configuration in the *Security Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Create a block source parameter profile and enter its view.

inspect block-source parameter-profile *parameter-name*

3. Set the block period during which a source IP address is blocked.

   block-period *period*

   The default setting is 1800 seconds.

Configuring a capture parameter profile

About this task

A capture parameter profile defines the following parameters for the capture action in DPI service modules:

- Maximum number of bytes that can be cached.
- Daily export time for cached packets.
- URL to which cached packets are exported (for example, tftp://192.168.100.100/upload).

The device caches captured packets locally and exports the cached packets to the designated URL at the daily export time or when the number of cached bytes reaches the limit. After the export, the device clears the local cache and starts to capture new packets.

Procedure

1. Enter system view.

   system-view

2. Create a capture parameter profile and enter its view.

   inspect capture parameter-profile *parameter-name*

3. Set the maximum volume of captured packets that can be cached.

   capture-limit *kilobytes*

   By default, the device can cache a maximum of 512 Kilobytes of captured packets.

4. Set the daily export time for cached captured packets.

   export repeating-at *time*

   By default, the cached captured packets are exported at 1:00 a.m. every day.

5. Specify the URL to which cached captured packets are exported

   export url *url-string*

   By default, no URL is specified for exporting the cached captured packets.

Configuring a logging parameter profile

About this task

A logging parameter profile defines the log output method and log output language for the logging action in DPI service modules.

Restrictions and guidelines

After setting the IPS log language to Chinese, only the attack name field of the IPS logs supports displaying in Chinese.

Procedure

1. Enter system view.

   system-view

2. Create a logging parameter profile and enter its view.

   inspect logging parameter-profile *parameter-name*

3. Specify the log export method.

   log { **email** | **syslog** }

   By default, logs are exported to the information center.

4. Set the language for IPS log output to Chinese.

   log language chinese

   By default, IPS logs are output in English.

Configuring a redirect parameter profile

About this task

A redirect parameter profile defines the URL to which packets are redirected for the redirect action in DPI service modules.

Restrictions and guidelines

The defined URL must start with http:// or https://. for example, http://www.example.com.

Procedure

1. Enter system view.

   system-view

2. Create a redirect parameter profile and enter its view.

   inspect redirect parameter-profile *parameter-name*

3. Specify the URL to which packets are redirected.

   redirect-url *url-string*

   By default, no URL is specified for packet redirecting.

Configuring an email parameter profile

About this task

An email parameter profile defines the following parameters for the email action in DPI service modules:

- Email server.
- Email sender and receiver.
- Username and password for logging in to the email server.

Procedure

1. Enter system view.

   system-view

2. Create an email parameter profile and enter its view.

   inspect email parameter-profile *parameter-name*

3. Specify the email server.

   email-server *addr-string*

   By default, no email server is specified.

4. Specify the email sender address.

   sender *addr-string*

   By default, no email sender address is specified.

5. Specify the email receiver address.

receiver *addr-string*

By default, no email receiver address is specified.

6. (Optional.) Configure email client authentication.
   a. Enable email client authentication.
      authentication enable
      By default, email client authentication is enabled.
   b. Specify the username for logging in to the email server.
      username *name-string*
      By default, no username is specified for logging in to the email server.
   c. Specify the password for logging in to the email server.
      password { **cipher** | **simple** } *string*
      By default, no password is specified for logging in to the email server.
   d. Enable the secure password transmission feature.
      secure-authentication enable
      By default, the secure password transmission feature is disabled.
7. Configure output limit for log entries sent to the email server.
   email-limit interval *interval* **max-number** *value*
   By default, the device allows sending a maximum of 10 log entries within five minutes.

Configuring a warning parameter profile

About this task

A warning parameter profile defines the parameters for the warning action in DPI service modules. After you create a warning parameter profile, you can import a user-defined alarm message from a file.

Procedure

1. Enter system view.
   system-view
2. Create a warning parameter profile and enter its view,.
   inspect warning parameter-profile *profile-name*
3. Import a user-defined alarm message from a warning file.
   import block warning-file *file-path*
   By default, the device uses the alarm message "**The site you are accessing has a security risk and thereby is blocked**."
4. (Optional.) Restore the default alarm message.
   reset block warning-file
   This command clears the user-defined alarm message and restores the default alarm message.

Optimizing the DPI engine

About this task

The DPI engine includes a series of optimization features. For example, you can enable the DPI engine to uncompress or decode the compressed or encoded packets to identify the application information of the packets. The optimization features improve inspection and accuracy of the DPI engine, but consume more system resources.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of payload-carrying packets to be inspected per data flow.

   inspect packet maximum *max-number*

   By default, the DPI engine can inspect a maximum of 32 payload-carrying packets per data flow.

3. Set the maximum number of options to be cached per TCP or UDP data flow.

   inspect cache-option maximum *max-number*

   By default, the DPI engine can cache a maximum of 32 options per TCP or UDP data flow.

4. Configure the TCP segment reassembly feature.

   - Enable TCP segment reassembly.

     inspect tcp-reassemble enable

     By default, the TCP segment reassembly feature is disabled.

   - Set the maximum number of TCP segments that can be cached for reassembly per TCP flow.

     inspect tcp-reassemble max-segment *max-number*

     By default, a maximum of 10 TCP segments can be cached for reassembly per TCP flow.

5. (Optional.) Disable a DPI engine optimization feature.

   inspect optimization [**chunk** | **no-acsignature** | **raw** | **uncompress** | **url-normalization**] **disable**

   By default, all DPI engine optimization features are enabled.

   You can disable DPI engine optimization features to improve the device performance as needed.

Enabling inspection suspension upon excessive CPU usage

About this task

Packet inspection of the DPI engine is a complex and resource-consuming process.

Inspection suspension upon excessive CPU usage works as follows:

- When the device's CPU usage rises to or above the CPU usage threshold, the DPI engine suspends packet inspection to guarantee the device performance.
- When the device's CPU usage drops to or below the CPU usage recovery threshold, the DPI engine resumes packet inspection.

For information about configuring the CPU usage thresholds, see device management in *Fundamentals Configuration Guide*.

Restrictions and guidelines

Do not disable inspection suspension upon excessive CPU usage if the device's CPU usage is high.

When the device's CPU usage is low, you can disable this feature to improve inspection accuracy.

Procedure

1. Enter system view.

```
system-view
```

2. Enable inspection suspension upon excessive CPU usage.

```
undo inspect cpu-threshold disable
```

By default, inspection suspension upon excessive CPU usage is enabled.

Configuring DPI engine inspection parameters

Configuring an inspection mode

About this task

Select an inspection mode as required:

- **Balanced mode**—Applicable to most scenarios. This mode makes a tradeoff between the device performance and inspection coverage. The maximum length is 32 Kilobytes for FTP, HTTP, SMB, NFS, and email streams, and the maximum file length for MD5 inspection is 2048 Kilobytes.
- **Large coverage mode**—Applicable to the scenarios that require large inspection coverage. This mode improves the inspection coverage at the cost of device performance. The maximum length is 128 Kilobytes for FTP, HTTP, SMB, NFS, and email streams, and the maximum file length for MD5 inspection is 5120 Kilobytes.
- **High performance mode**—Applicable to the scenarios that require high device performance. This mode improves the device performance while ensuring a certain inspection coverage. The maximum length is 32 Kilobytes for FTP, HTTP, SMB, NFS, and email streams, and the maximum file length for MD5 inspection is 32 Kilobytes.
- **User-defined mode**—Applicable to the scenarios that have specific requirements for inspection coverage and device performance. In this mode, you can execute the **inspect stream-fixed-length and inspect md5-fixed-length** commands **to** set the maximum stream length for inspection and maximum file length for MD5 value calculation, respectively.

The maximum lengths for stream inspection and MD5 inspection will not change after the user-defined inspection mode is switched, and you can adjust those lengths as required.

Procedure

1. Enter system view.

```
system-view
```

2. Configure a DPI engine inspection mode.

```
inspect coverage { balanced | large-coverage | high-performance |
user-defined }
```

By default, the DPI engine uses the balanced mode.

Configuring stream fixed length inspection

About this task

DPI engine inspects only the fixed-length data after the first packet for each stream. The remaining stream data is not inspected. Reducing the fixed length for stream inspection enhances the inspection efficiency.

Restrictions and guidelines

This feature can be configured only if the DPI engine inspection mode is user-defined mode.

Procedure

1. Enter system view.

   system-view

2. Enable the stream fixed length inspection.

   undo inspect stream-fixed-length disable

   By default, the stream fixed length inspection is enabled.

3. Set the fixed length for stream inspection.

   inspect stream-fixed-length { **email** | **ftp** | **http** | **nfs** | **smb** } * *length*

   The default length is 32 Kilobytes for FTP, HTTP, NFS, SMB, and email streams.

   The longer the inspection data length, the lower the device throughput, and the higher the packet inspection accuracy.

Configuring file fixed length inspection

About this task

DPI engine inspects only the fixed-length data of files in each data stream. The remaining data of the file is not inspected. This is because virus signatures are typically embedded in the first half of a file. Narrowing the inspection scope enhances the file inspection efficiency.

Restrictions and guidelines

This feature can be configured only if the DPI engine inspection mode is user-defined mode.

Because files are transmitted in a data stream, the fixed length of files must not be longer than that of the data stream.

Procedure

1. Enter system view.

   system-view

2. Enable the file fixed length inspection.

   inspect file-fixed-length enable

   By default, the file fixed length inspection is disabled.

3. Set the fixed length for file inspection.

   inspect file-fixed-length { **email** | **ftp** | **http** | **nfs** | **smb** } * *length-value*

   By default, the fixed length is 32 Kilobytes for FTP, HTTP, NFS, SMB, and email files.

   If a data stream contains multiple files, this feature inspects only the fixed length data of each file.

Configuring MD5 fixed-length file inspection

About this task

In the anti-virus services, the device matches the packet signatures, calculates MD5 values of the files, and compares the calculated MD5 values with the MD5 rules in the signature library. If the MD5 value for a file matches an MD5 rule in the signature library, the file is considered to contain viruses. For more information about virus inspection, see "Configuring anti-virus."

The DPI engine inspects the packet signatures and MD5 values at the same time. After the length of a file reaches the fixed stream inspection length, the DPI engine will stop the packet signature inspection. To continue the MD5 inspection for a file, enable MD5 fixed-length file inspection and configure a fixed file length for MD5 inspection longer than the fixed length for stream inspection. Then the DPI engine calculates MD5 value of the fixed-length file.

The increase of the file length for MD5 inspection will reduce the device performance but improve the success rate of the MD5 inspection. The decrease of the file length for MD5 inspection will improve the device performance but reduce the success rate of the MD5 inspection.

Restrictions and guidelines

This feature can be configured only if the DPI engine inspection mode is user-defined mode.

Procedure

1. Enter system view.

   `system-view`

2. Enable MD5 fixed-length file inspection.

   `inspect md5-fixed-length enable`

   By default, MD5 fixed-length file inspection is enabled.

3. Set the fixed file length for MD5 inspection.

   `inspect md5-fixed-length` { **`email`** | **`ftp`** | **`http`** | **`nfs`** | **`smb`** } * *`length`*

   By default, the fixed length of FTP, HTTP, SMB, NFS, and email files for MD5 inspection is 2048 Kilobytes.

Configuring MD5 hash-based virus inspection for all files

Restrictions and guidelines

This feature might degrade the processing performance of other services. Enable it only when necessary.

Procedure

1. Enter system view.

   `system-view`

2. Enable MD5 hash-based virus inspection for all files.

   `inspect md5-verify all-files`

   By default, the DPI engine performs MD5 hash-based virus inspection only for executable files, office files, and compressed files.

Setting the maximum data size for file decompression

About this task

The device can decompress .zip files for file data inspection. Perform this task to set the maximum data size that can be decompressed in a file. The remaining file data will be ignored.

Small limits might make DPI engine unable to identify the original file content correctly, reducing the impact on the device forwarding performance but affecting the accuracy of the file inspection results for DPI services (such as anti-virus and data filtering).

Restrictions and guidelines

The device can decompress only .zip files.

Procedure

1. Enter system view.

   `system-view`

2. Set the maximum data size that can be decompressed in a .zip file.

   inspect file-uncompr-len *max-size*

   By default, the device can decompress a maximum of 100 MB data in a .zip file.

Setting the maximum number of file decompression operations

About this task

The DPI engine consumes memory resources each time it performs a file decompression operation. A large number of file decompression operations might consume a large number of memory resources. Perform this task to limit the memory resources consumed by file decompression operations.

A small limit can reduce memory consumption but might reduce the detection success rate of the DPI engine. A great limit might improve the detection success rate of the DPI engine but degrade the device performance.

Restrictions and guidelines

This feature is supported only on the default context. For more information about contexts, see *Virtual Technologies Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of file decompression operations.

   inspect uncompress maximum *max-number*

   By default, the maximum number of file decompression operations is calculated according to the actual memory size of the device.

Setting the maximum number of NFS file names recorded

About this task

The DPI engine records file names during file detection for users to obtain file information in logs. The record process occupies memory resources. The more files detected, the more memory resources occupied. In an environment using NFS to transfer a large number of files, perform this task to limit the memory resources consumed by recording file names.

In scenarios requiring high performance, you can set a small limit to reduce memory consumption. In scenarios not requiring high performance, you can set a great limit to enable users to obtain more file information.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of NFS file names recorded.

   inspect record-filename nfs maximum *max-number*

   By default, the maximum number of NFS file names recorded is calculated according to the actual memory size of the device.

Configuring advanced features of the DPI engine

Enabling source port-based application identification

About this task

You can use this feature to identify traffic of applications that use fixed source ports when the following conditions are true:

- The types of traffic transmitted over networks are relatively unvaried and use fixed source ports.
- Destination port-based application identification or signature-based traffic content identification is not supported.

The application identification results produced by this feature might not be accurate. Configure this feature according to your live network as a best practice.

Procedure

1. Enter system view.

   `system-view`

2. Enable source port-based application identification.

   `inspect source-port-identify enable`

   By default, source port-based application identification is disabled.

Specifying a proxy server for online DPI service signature update

About this task

The device must access the company's website for online signature update of DPI services (such as URL filtering). If direct connectivity is not available, the device can access the company's website through the specified proxy server. For more information about online signature update, see "Configuring URL filtering" and "Configuring anti-virus."

Restrictions and guidelines

If you specify a proxy server by domain name instead of IP address, make sure the device can resolve the domain name into an IP address through DNS. For more information about DNS, see *Layer 3—IP Services Configuration Guide.*

Procedure

1. Enter system view.

   `system-view`

2. Specify a proxy server for online DPI service signature update

   inspect signature auto-update proxy { **domain** *domain-name* | **ip** *ip-address* } [**port** *port-number*] [**user** *user-name* **password** { **cipher** | **simple** } *string*]

   By default, the proxy server used by DPI services for online signature update is not specified.

Specifying the cloud query server for DPI services

About this task

You can specify the server used for cloud query by DPI services.

Currently, the cloud query server supports URL filtering cloud query and anti-virus MD5 value cloud query.

Restrictions and guidelines

For successful cloud query, make sure the device can resolve the host name of the cloud query server into an IP address through DNS. For more information about DNS, see DNS configuration in *Layer 3—IP Services Configuration Guide.*

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify the cloud query server for DPI services.

   ```
   inspect cloud-server host-name
   ```

Enabling support for HA dual-active mode

About this task

The feature ensures the device in HA dual-active mode can correctly process DPI services in a network with asymmetric forwarding of flows.

For more information about HA dual-active mode, see RBM-based hot backup configuration in *High Availability Configuration Guide*.

Restrictions and guidelines

This feature takes effect only when the device operates in HA dual-active mode.

This feature might degrade device performance.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable support for HA dual-active mode.

   ```
   inspect dual-active enable
   ```

   By default, support for HA dual-active mode is disabled.

Configuring real source IP inspection

Enabling real source IP inspection

About this task

When a client connects to a Web server through HTTP proxies, the source IP address of the request will change. To identify the source IP attacks accurately, you can enable real source IP inspection to obtain the real source IP address from the corresponding fields in the request.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable real source IP inspection.

   ```
   inspect real-ip enable
   ```

   By default, real source IP inspection is disabled.

Setting the priority of an inspected field for real source IP inspection

About this task

With real source IP inspection enabled, the device obtains the real source IP address of the client by inspecting multiple fields in the packets by default.

When multiple IP addresses are detected, the devices uses the IP address obtained from the field with the highest priority as the final real source IP address.

Procedure

1. Enter system view.

   `system-view`

2. Set the priority of an inspected field for real source IP inspection.

   `inspect real-ip detect-field` { **`cdn-src-ip`** | **`tcp-option`** | **`x-real-ip`** | **`xff`** } **`priority`** *`priority-value`*

   By default, no priority is specified for any inspected field in the real source IP inspection, and all inspected fields use priority value 0. The device inspects the fields in the order of the **`xff`**, **`cdn-src-ip`**, **`x-real-ip`**, and **`tcp-option`** fields.

Configuring real source IP address inspection for the X-Forwarded-For field

About this task

When a client connects to a Web server through an HTTP proxy, the HTTP header might contain the X-Forwarded-For field that carries multiple IP addresses. The standard syntax of the X-Forwarded-For field is <client>, <proxy1>, <proxy2>,…<proxyn>. If a request goes through multiple proxies, the IP addresses of each successive proxy are listed. The rightmost IP address is the IP address of the most recent proxy and the leftmost IP address is the IP address of the originating client.

Procedure

1. Enter system view.

   `system-view`

2. Configure real source IP address inspection for the X-Forwarded-For field.

   `inspect real-ip detect-field xff` { **`head`** | **`tail`** }

   By default, the rightmost IP address in the X-Forwarded-For field is the real source IP address.

Configuring real source IP inspection for the TCP Options field

About this task

To enable the device to locate the real source IP address in the TCP Option field, you must first define a hexadecimal string. If no hexadecimal string is found, the device will stop searching the TCP Options field for the real IP address.

Restrictions and guidelines

With real source IP inspection enabled, the device does not obtain the real source IP address from the TCP Options field by default. The device searches the real source IP from the TCP Options field only after the parameters are configured.

Procedure

1. Enter system view.

   `system-view`

2. Configure real source IP inspection for the TCP Options field.

   inspect real-ip detect-field tcp-option hex *hex-vector* [**offset** *offset-value*] [**depth** *depth-value*] [**ip-offset** *ip-offset-value*]

   By default, real source IP inspection is not configured for the TCP Options field, and the device does not obtain the real source IP address from the TCP Options field.

Disabling the DPI engine

Disabling the DPI engine for all protocols

About this task

Packet inspection in the DPI engine is a complex and resource-consuming process. When the CPU usage is too high, you can disable the DPI engine to guarantee the device performance.

Procedure

1. Enter system view.

   `system-view`

2. Disable the DPI engine.

   `inspect bypass`

   By default, the DPI engine is enabled.

⚠ **CAUTION:**

This command causes packets of any protocols not to be processed by DPI. DPI-based services might also be interrupted. For example, security policies cannot control access to applications.

Disabling the DPI engine for the specified protocols

About this task

Perform this task in the following scenarios:

- Scenario 1: Inspection on packets of the specified protocols is not required. You can disable the DPI engine for the specified protocols to reduce the consumption of device resources and improve the device performance.
- Scenario 2: Inspection on packets of the specified protocols causes device reboot. You can specify the protocols to bypass the DPI engine to avoid device reboot caused by inspection error and ensure the inspection on packets of other protocols.

To disable the DPI engine for the specified protocols, you can use either of the following methods:

- **Manual configuration**—If the administrator knows the protocols to bypass, you can use this method. This method applies to scenario 1.

- **Automatic configuration**—This method applies to scenario 2. If you use this method, the device automatically identifies the protocols to bypass the DPI engine after device reboot.

Procedure

1. Enter system view.

   system-view

2. Disable the DPI engine for the specified protocols.
   - Manually disable the DPI engine for the specified protocols.

     inspect bypass protocol { **dns** | **ftp** | **ftp-data** | **http** | **https** | **imap** | **nfs** | **pop3** | **rtmp** | **sip** | **smb** | **smtp** | **telnet** | **tftp** } *

     By default, the DPI engine inspects all supported protocols.
   - Automatically disable the DPI engine for the specified protocols.

     inspect auto-bypass enable

     By default, automatic bypass of the DPI engine is disabled.

Display and maintenance commands for DPI engine

Execute **display** commands in any view.

Task	Command
Display the status of the DPI engine.	**display inspect status**
Display information about the MD5 hash-based virus inspection for all files feature.	**display inspect md5-verify configuration**

Contents

Configuring IPS

About IPS

Intrusion prevention system (IPS) is a security feature that enables devices to monitor network traffic for malicious activity and to proactively take prevention actions.

IPS functions

IPS provides the following functions:

- **In-depth protection**—IPS inspects the application layer data of packets, performs protocol analysis and reassembly on network traffic flows, and takes actions according to the analysis results.
- **Real-time protection**—IPS monitors network traffic in real-time and can take actions on detected attacks.
- **All-around protection**—IPS can detect and prevent the following types of attacks:
  - Malicious software such as worms, viruses, Trojan, bots, spyware, adware, scanners, and backdoors.
  - Malicious attacks such as common gateway interface (CGI) attacks, cross-site scripting attacks, injection attacks, directory traversal attacks, information leakage attacks, remote file inclusion attacks, buffer overflow attacks, code execution attacks, and DoS attacks.
- **Bidirectional protection**—IPS monitors both incoming and outgoing traffic to prevent attacks arising from the internal and external networks.

IPS policies

IPS is implemented based on IPS policies. An IPS policy contains a set of IPS signatures for matching packets and the actions for the packets.

IPS signatures

The device compares packets with IPS signatures to detect, classify, and prevent network attacks.

Each IPS signature contains various attributes, including attack category, action, protected target, severity level, and direction. You can filter the IPS signatures that an IPS policy uses based on the IPS signature attributes.

The device supports the following types of IPS signatures:

- **Predefined IPS signatures**—Automatically generated by the device based on the local signature library. You cannot add, modify, or delete a predefined IPS signature.
- **User-defined IPS signatures**—For new attacks that cannot be detected by predefined signatures, you can customize IPS signatures. The user-defined IPS signatures include Snort signatures that are imported from a Snort file and user-configured signatures that are manually configured.

IPS actions

When the device detects a matching packet for an IPS signature, it takes the actions specified for the signature on the packet.

The device supports the following signature actions:

- **Reset**—Closes the TCP connections for matching packets by sending TCP reset messages.
- **Redirect**—Redirects matching packets to a webpage.

- **Block-source**—Drops matching packets and adds the sources of the packets to the IP blacklist. If the IP blacklist feature is enabled, packets from the blacklisted sources will be blocked for a duration set by the **`block-period`** command. If the IP blacklist feature is not enabled, packets from the blacklisted sources are not blocked.

  To enable the IP blacklist feature, use the **`blacklist global enable`** command. For more information about the IP blacklist feature, see *Security Configuration Guide*.

  For more information about the **`block-period`** command, see *DPI Command Reference*.
- **Drop**—Drops matching packets.
- **Permit**—Permits matching packets to pass.
- **Capture**—Captures matching packets.
- **Logging**—Logs matching packets.

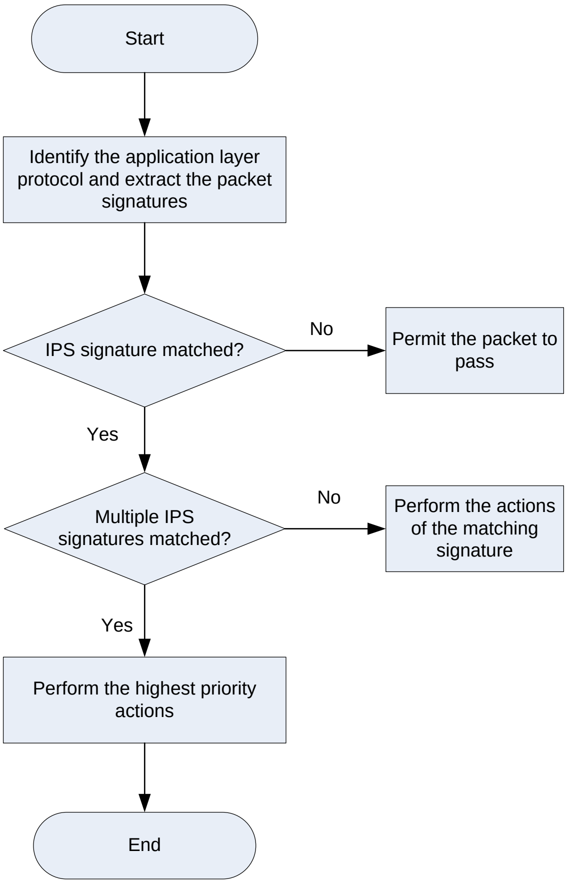
IPS mechanism

IPS takes effect after you apply an IPS policy to a DPI application profile and use the DPI application profile in a security policy rule.

As shown in Figure 1, upon receiving a packet, the device performs the following operations:

1. The device identifies the packet application layer protocol and extracts the packet signatures.
2. The device determines the actions for the packet by comparing the extracted packet signatures with the IPS signatures in the IPS policy:
   - If the packet does not match any IPS signatures, the device permits the packet to pass.
   - If the packet matches only one IPS signature, the device takes the signature actions.
   - If the packet matches multiple IPS signatures, the device uses the following rules to select the actions:
     - If the matching IPS signatures have two or more actions, including **redirect**, **drop, permit**, and **reset**, the device takes the action of the highest priority. The actions in descending order of priority are **reset**, **redirect**, **drop**, and **permit**.
     - The device will execute the **block-source**, **capture**, and **logging** actions if they are in the matching IPS signatures.

Figure 1 IPS mechanism


Start
Identify the application layer protocol and extract the packet signatures
IPS signature matched?
No
Permit the packet to pass
Yes
Multiple IPS signatures matched?
No
Perform the actions of the matching signature
Yes
Perform the highest priority actions
End

IPS signature library management

The device uses IPS signatures to inspect application layer traffic for malicious threats and attacks.

You can update the device IPS signature library to the latest version or roll back the library to the previous or the factory default version.

Updating the IPS signature library

The following methods are available for updating the IPS signature library on the device:

- Automatic update.

  The device automatically downloads the most up-to-date IPS signature file to update its local signature library periodically.
- Triggered update.

  The device downloads the most up-to-date IPS signature file to update its local signature library immediately after you trigger the operation.
- Manual update.

  Use this method when the device cannot obtain the IPS signature file automatically.

  You must manually download the most up-to-date IPS signature file, and then use the file to update the signature library on the device.

Rolling back the IPS signature library

If filtering false alarms or filtering exceptions occur frequently, you can roll back the IPS signature library to the previous version or to the factory default version.

Restrictions: Licensing requirements for IPS

The IPS module requires a license to run on the device. If the license expires, you can still use the IPS functions but you can no longer upgrade the IPS signature library on the device. For more information about licenses, see license management in *Fundamentals Configuration Guide*.

IPS tasks at a glance

To configure IPS, perform the following tasks:

1. Configuring an IPS policy
2. Applying an IPS policy to a DPI application profile
3. (Optional.) Activating IPS policy settings
4. Applying a DPI application profile to a security policy rule
5. Managing the IPS signature library
6. (Optional.) Importing and deleting Snort IPS signatures
7. (Optional.) Managing a user-configured IPS signature
8. (Optional.) Enabling IPS signature hit counting
9. (Optional.) Configuring IPS whitelist

Configuring an IPS policy

Creating an IPS policy

About this task

By default, a newly created IPS policy uses all enabled IPS signatures and applies to the packet matching a signature the default signature action. You can filter the IPS signatures used by the IPS policy and change the signature actions.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Create an IPS policy and enter its view.

   ```
   ips policy policy-name
   ```

   A default IPS policy named **default** exists. The default IPS policy uses all enabled IPS signatures on the device and cannot be modified or deleted.

Configuring IPS signature filtering criteria for an IPS policy

About this task

By default, an IPS policy uses all enabled IPS signatures on the device. You can set criteria to filter IPS signatures that an IPS policy uses based on the signature attributes.

An IPS policy uses an IPS signature only if the signature matches all the configured criteria.

For certain attribute-based criterion (such as the action, object direction, or severity level criterion), you can specify multiple attribute values. An IPS signature matches the criterion if it matches any of the specified attribute values.

Procedure

1. Enter system view.

   system-view

2. Enter IPS policy view.

   ips policy *policy-name*

3. Configure the IPS signature filtering criteria.

   - Set a target criterion.

     protect-target { *target* [*subtarget* | **all**] }

     By default, the target attribute is not used for IPS signature filtering.

   - Set an attack category criterion.

     attack-category { *category* [*subcategory*] | **all** }

     By default, the attack category attribute is not used for IPS signature filtering.

   - Set an action criterion.

     action { **block-source** | **drop** | **permit** | **reset** } *

     By default, the action attribute is not used for IPS signature filtering.

   - Set an object direction criterion.

     object-dir { **client** | **server** } *

     By default, the object direction attribute is not used for IPS signature filtering.

   - Set a severity level criterion.

     severity-level { **critical** | **high** | **low** | **medium** } *

     By default, the severity level attribute is not used for IPS signature filtering.

   - Set a default status criterion.

     status { **disabled** | **enabled** } *

     By default, the default status attribute is not used for IPS signature filtering.

     In the IPS signature library, the default status of an IPS signature indicates whether or not the IPS signature is recommended. IPS signatures in **disabled** default status are not recommended, and IPS signatures in **enabled** default status are recommended.

Specifying an IPS signature library baseline version

About this task

This feature sets an IPS signature library version as the baseline version and enables the device to match packets only with the signatures in the baseline version. With this feature, the device compares the current IPS signature library with the baseline signature library. If a signature is included in the current signature library but does not included in the baseline signature library, the device sets the signature to ineffective state. Signatures in ineffective state cannot match packets.

This feature allows the device to match packets only with the signatures in the baseline version without rolling back the signature library to the baseline version.

To separately activate an ineffective signature after this feature is used, perform the following tasks:

1. On the Web interface of the device, obtain the IDs of all ineffective signatures.
2. Use this feature again to change the IPS signature library baseline version to the version that contains the signature.
3. Execute the **signature override** command to disable all signatures that were in ineffective state when the previous signature library baseline version was used, except the signature to be activated.

Procedure

1. Enter system view.

   system-view

2. Enter IPS policy view.

   ips policy *policy-name*

3. Specify an IPS signature library baseline version.

   signature version-baseline *version-number*

   By default, no IPS signature library version baseline is configured.

Configuring IPS actions for an IPS policy

About this task

By default, the system applies the default actions of an IPS signature to packets matching the signature.

You can also configure global actions for an IPS policy or change the actions for individual IPS signatures in the policy.

The system selects the actions for packets matching an IPS signature in the following order:

1. Actions configured for the IPS signature in the IPS policy.
2. Actions configured for the IPS policy.
3. Default actions of the IPS signature.

Restrictions and guidelines

The **logging** keyword enables the IPS module to log packet matching events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output IPS logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view IPS logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

In IPS logs, the values for the real source MAC address (RealSrcMacAddr) and real destination MAC address (RealDstMacAddr) are displayed only when MAC learning through a Layer 3 device is enabled. For more information about MAC learning through a Layer 3 device, see *Fundamentals Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an IPS policy.

   ips policy *policy-name*

3. Specify the global packet processing actions for the IPS policy.

   signature override all { { **block-source** | **drop** | **permit** | **redirect** | **reset** } | **capture** | **logging** } *

By default, no actions are specified for an IPS policy and the default actions of IPS signatures are applied to matching packets.

4. (Optional.) Change the state or actions for an IPS signature.

   signature override { **pre-defined** | **user-defined** } *signature-id* { { **disable** | **enable** } [{ **block-source** | **drop** | **permit** | **redirect** | **reset** } | **capture** | **logging**] * }

   By default:

   - Predefined IPS signatures use the actions and states defined by the system.
   - User-defined IPS signatures use the actions and states defined in the IPS signature file from which the signatures are imported.

   You cannot change the state or actions for an IPS signature in the default IPS policy.

5. (Optional.) Specify the number of the captured packets to be cached for threat analysis.

   ips capture-cache *number*

   By default, the number of the captured packets to be cached for threat analysis is not specified, and the device does not cache any captured packets.

   This command enables the device to cache the IPS captured packets. After caching the specified number of the captured packets, the device writes all cached packets and the hit packet into the capture file for threat analysis.

Specifying a parameter profile for an IPS action

About this task

The **block source**, **capture**, and **logging** actions take effect only after a parameter profile is specified. You can specify a parameter profile for an IPS action as follows:

- Specify a global parameter profile in system view. The setting takes effect in all IPS policies.
- Specify a parameter profile in IPS policy view, which a policy-specific setting. Only the email action supports specifying a parameter profile in IPS policy view.

Restrictions and guidelines

- The global parameter profile for an IPS action takes precedence over a policy-specific parameter profile for the action.
- To have a parameter profile for an IPS action in an IPS policy take effect, make sure the global parameter profile is disabled.
- As a best practice, enable the global parameter profile after the global parameter profile configuration is completed.

Specifying a global parameter profile for an IPS action

1. Enter system view.

   system-view

2. Specify a global parameter profile for an IPS action.

   ips { **block-source** | **capture** | **email** | **logging** | **redirect** } **parameter-profile** *parameter-name*

   By default, no global parameter profile is specified for an IPS action.

   If you do not specify a parameter profile for an action, or if the specified profile does not exist, the default action parameter settings are used. For more information about configuring an action parameter profile, see DPI engine commands in *DPI Command Reference*.

Specifying an action parameter profile in an IPS policy

1. Enter system view.

system-view

2. Enter IPS policy view.

   ips policy *policy-name*

3. Specify the log output method.

   log { **email** | **syslog** }

   By default, the IPS log output method is **syslog**.

4. (Optional.) Specify the lowest severity level of the matching IPS signatures for log output via email.

   email severity-level { **critical** | **high** | **low** | **medium** }

   By default, the lowest severity level of the matching IPS signatures for log output via email is **low**.

   This command is available only when the log output method is **email**. The system sends emails for IPS logs only when the severity levels of the matching IPS signatures are not lower than specified severity level.

5. (Optional.) Specify a parameter profile for the email action.

   email parameter-profile *parameter-profile-name*

   By default, no parameter profile is specified for the email action.

   This command is available only when the log output method is **email**. For more information about configuring an email parameter profile, see DPI engine commands in *DPI Command Reference*.

6. (Optional.) Disable the global parameter profile.

   undo global-parameter enable

   By default, global parameter profiles are enabled.

Applying an IPS policy to a DPI application profile

About this task

An IPS policy must be applied to a DPI application profile to take effect.

Restrictions and guidelines

A DPI application profile can use only one IPS policy. If you apply different IPS policies to the same DPI application profile, only the most recent configuration takes effect.

Procedure

1. Enter system view.

   system-view

2. Enter DPI application profile view.

   app-profile *profile-name*

   For more information about this command, see DPI engine commands in *DPI Command Reference*.

3. Apply an IPS policy to the DPI application profile.

   ips apply policy *policy-name* **mode** { **protect** | **alert** }

   By default, no IPS policy is applied to the DPI application profile.

Activating IPS policy settings

About this task

By default, when a configuration change (such as creation, modification, or deletion) to an IPS policy or rule occurs, the system starts to detect configuration changes at 20-second intervals.

- If no configuration change occurs within 20 seconds after this change, the system will perform an activation operation after the next 20 seconds.
- If a new configuration change occurs within 20 seconds after this change, the system continues to detect configuration changes at 20-second intervals.

To immediately activate a configuration change, execute the **`inspect activate`** command.

For more information about activating DPI service module configuration, see "Configuring the DPI engine."

Procedure

1. Enter system view.
   `system-view`
2. Activate IPS policy settings.
   `inspect activate`
   By default, the system automatically activates changed IPS policy and rule settings for them to take effect.

△ **CAUTION:**

This command can cause temporary outage for DPI services. Services based on the DPI services might also be interrupted. For example, security policies cannot control application access.

Applying a DPI application profile to a security policy rule

1. Enter system view.
   `system-view`
2. Enter security policy view.
   `security-policy` { **`ip`** | **`ipv6`** }
3. Enter security policy rule view.
   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }
4. Set the rule action to pass.
   `action pass`
   The default rule action is **`drop`**.
5. Use a DPI application profile in the rule.
   `profile` *`app-profile-name`*
   By default, no DPI application profile is used in a security policy rule.

Managing the IPS signature library

You can update or roll back the version of the IPS signature library on the device.

Restrictions and guidelines

- Do not delete the **/dpi/** folder in the root directory of the storage medium.
- Do not perform IPS signature update or rollback when the device's free memory is below the normal state threshold. For more information about device memory thresholds, see device management in *Fundamentals Configuration Guide*.
- For successful automatic and immediate signature update, make sure the device can resolve the domain name of the company's website into an IP address through DNS. For more information about DNS, see DNS configuration in *Layer 3—IP Services Configuration Guide.*
- Update only one signature library at a time. Do not perform signature library update until the existing signature library update is completed.

Scheduling automatic IPS signature library update

About this task

You can schedule automatic IPS signature library update if the device can access the signature database services on the company's website. The device periodically obtains the latest signature file from the company's website to update its local signature library according to the update schedule.

Procedure

1. Enter system view.

   `system-view`

2. Enable automatic IPS signature library update and enter automatic IPS signature library update configuration view.

   `ips signature auto-update`

   By default, automatic IPS signature library update is disabled.

3. Schedule the update time.

   `update schedule` { **`daily`** | **`weekly`** { **`fri`** | **`mon`** | **`sat`** | **`sun`** | **`thu`** | **`tue`** | **`wed`** } } **`start-time`** *`time`* **`tingle`** *`minutes`*

   By default, the device updates the IPS signature library at a random time between 01:00:00 and 03:00:00 every day.

4. (Optional.) Configure the device to overwrite the current IPS signature library without backing up the library during an automatic signature library update.

   `override-current`

   By default, the device backs up the current IPS signature library as the previous version before performing an automatic IPS signature library update.

Triggering an immediate IPS signature update

About this task

Anytime you find a release of new signature version on the company's website, you can trigger the device to immediately update the local signature library.

Procedure

1. Enter system view.

```
system-view
```

2. Trigger an immediate IPS signature library update.

```
ips signature auto-update-now
```

Performing an IPS signature manual update

About this task

If the device cannot access the signature database services on the company's website, use one of the following methods to manually update the IPS signature library on the device:

- **Local update**—Updates the IPS signature library by using a locally stored update IPS signature file.

  Store the update file on the master device for successful signature library update.
- **FTP/TFTP update**—Updates the IPS signature library by using the file stored on the FTP or TFTP server.

Procedure

1. Enter system view.

```
system-view
```

2. Manually update the IPS signature library on the device.

```
ips signature update [ override-current ] file-path [ vpn-instance vpn-instance-name ]
```

Rolling back the IPS signature library

About this task

If an IPS signature library update causes exceptions or a high false alarm rate, you can roll back the IPS signature library.

Before rolling back the IPS signature library, the device backs up the current signature library as the previous version. For example, the previous library version is V1 and the current library version is V2. If you perform a rollback to the previous version, library version V1 becomes the current version and library version V2 becomes the previous version. If you perform a rollback to the previous version again, the library rolls back to library version V2.

Procedure

1. Enter system view.

```
system-view
```

2. Roll back the IPS signature library to the previous version or to the factory default version.

```
ips signature rollback { factory | last }
```

Enabling logging for IPS signature library update and rollback events

About this task

This feature enables logging for successful IPS signature library update and rollback events and outputs the logs at the specified daily time.

Restrictions and guidelines

The device supports outputting IPS signature library update and rollback logs only as fast logs to log hosts. For the IPS logs to be output correctly, make sure the following requirements are met:

- Fast log output of IPS logs in SGCC format are enabled by using the **`customlog format dpi ips sgcc`** command.
- The log hosts where the IPS logs should be sent are configured by using the **`customlog host`** command.

For more information about the preceding commands, see fast log output commands in *Network Management and Monitoring Command Reference*.

Procedure

1. Enter system view.

```
system-view
```

2. Enable IPS signature library update logging and set the daily log output time.

```
ips signature update-log send-time time
```

Importing and deleting Snort IPS signatures

Importing Snort IPS signatures

About this task

To add your own IPS signatures, create an IPS signature file in the Snort format and import the signatures from the file to the device.

Restrictions and guidelines

Make sure the IPS signature file contains all Snort signatures that you want to use. All existing Snort signatures on the device will be overwritten by the imported signatures.

For a signature defined by a Snort rule to be imported correctly from the IPS signature file, make sure Snort rule is valid.

Procedure

1. Enter system view.

```
system-view
```

2. Import Snort IPS signatures from a Snort file.

```
ips signature import snort file-path
```

Deleting Snort IPS signatures

1. Enter system view.

```
system-view
```

2. Delete all Snort IPS signatures.

```
ips signature remove snort
```

Managing a user-configured IPS signature

Creating a user-defined IPS signature

About this task

You can create signatures that do not exist in the current signature library.

Procedure

1. Enter system view.

   system-view
2. Create an IPS signature and enter its view.

   ips signature user-defined name *signature-name*

   By default, no user-configured IPS signatures exist.
3. (Optional.) Configure the description for the user-defined IPS signature.

   description *text*

Configuring attributes in a user-defined IPS signature

About this task

Each IPS signature contains various attributes, including action, direction, severity level, and the logical operator between the rules in the signature.

You can create multiple rules in a user-defined IPS signature. If the logical AND operator is specified, a packet matches the signature only when the packet matches all rules in the signature. If the logical OR operator is specified, a packet matches the signature when the packet matches any rule in the signature.

Procedure

1. Enter system view.

   system-view
2. Enter the view of a user-defined IPS signature.

   ips signature user-defined name *signature-name*

   By default, no user-defined IPS signatures exist.
3. Configure the attributes for the user-defined IPS signature.
   - Set the actions for packets matching the IPS signature.

     action { **block-source** | **drop** | **permit** | **reset** } [**capture** | **logging**] *

     By default, the action for a user-defined IPS signature is **permit**.
   - Set the traffic direction attribute.

     direction { **any** | **to-client** | **to-server** }

     By default, both client-to-server and server-to-client directions are defined for a user-defined IPS signature.
   - Set a severity level.

     severity-level { **critical** | **high** | **low** | **medium** } *

     By default, the low severity level attribute is specified for a user-defined IPS signature.
   - Set a logical operator between the rules in the signature.

     rule-logic { **and** | **or** }

By default, the logical OR operator is specified between the rules in a user-defined IPS signature.

Configuring rules for a user-defined IPS signature

About this task

A user-defined IPS signature rule can be one of the following types:

- **Keyword**.
- **Integer**.

A user-defined signature rule might contain filtering criteria, detection items, and a detection trigger condition. The device uses the rule for packet filtering as follows:

1. The device compares the packet with the filtering criteria.
   - If the packet matches all filtering criteria, the device goes to the next step.
   - If the packet does not match all filtering criteria, IPS does not process the packet.
2. The device compares the packet with the detection trigger condition.

   This step is available only for a rule of the keyword type.
   - If the packet matches the detection trigger condition, the device goes to the next step.
   - If the packet does not match the detection trigger condition, IPS does not process the packet.
3. The device compares the packet with the detection items.

   The detection items are used to match the specified contents in a packet. A packet matches a rule only when the packet matches all detection items in the rule. The match order of the detection items is their configuration order.

Restrictions and guidelines

A detection item compares its keyword with the contents in the specified protocol field.

To avoid detection errors, configure detection items based on the sequence of protocol fields in the HTTP protocol.

In a signature rule of the keyword match pattern type, a detection trigger condition must be configured before detection item configuration. If you delete the detection trigger condition, all detection items in the rule will also be deleted.

To define the start and end positions for the match operation, use either the offset and depth, or the relative offset and relative depth.

Procedure

1. Enter system view.

   `system-view`
2. Enter the view of a user-defined IPS signature.

   `ips signature user-defined name` *`signature-name`*
3. Create a user-defined IPS signature rule and enter its view.

   `rule` *`rule-id`* **`l4-protocol`** *`l4-protocol-name`* **`l5-protocol`** *`l5-protocol-name`* **`pattern-type`** { **`keyword`** | **`integer`** }

   By default, no user-defined IPS signature rules exist.
4. Configure the filtering criteria for the rule.
   - Set a source IP address filtering criterion.

     `source-address ip` *`ip-address`*

     By default, a user-defined IPS signature rule matches all source IP addresses.

- Set a destination IP address filtering criterion.

  destination-address ip *ip-address*

  By default, a user-defined IPS signature rule matches all destination IP addresses.
- Set source port filtering criteria.

  source-port *start-port* [**to** *end-port*]

  By default, a user-defined IPS signature rule matches all source ports.
- Set destination port filtering criteria.

  destination-port *start-port* [**to** *end-port*]

  By default, a user-defined IPS signature rule matches all destination ports.
- Set an HTTP request method filtering criterion.

  http-method *method-name*

  By default, a user-defined IPS signature rule matches all HTTP request methods.

5. Configure the detection trigger condition and detection items for a rule in a signature of the keyword type.

a. Create a detection trigger condition.

trigger field *field-name* **include** { **hex** *hex-string* | **text** *text-string* } [**offset** *offset-value*] [**depth** *depth-value*]

b. Create a detection item.

detection-keyword *detection-id* **field** *field-name* **match-type** { **exclude** | **include** } { **hex** *hex-string* | **regex** *regex-pattern* | **text** *text-string* } [**offset** *offset-value* [**depth** *depth-value*] | **relative-offset** *relative-offset-value* [**relative-depth** *relative-depth-value*]]

6. Configure detection items for a rule in a signature of the integer type.

detection-integer field *field-name* **match-type** { **eq** | **gt** | **gt-eq** | **lt** | **lt-eq** | **nequ** } *number*

Enabling IPS signature hit counting

About this task

This feature enables the device to collect hit statistics for each IPS signature. You can view IPS signature hit statistics on the Web interface of the device.

Procedure

1. Enter system view.

system-view

2. Enable the view of an IPS policy.

ips policy *policy-name*

3. Enable IPS signature hit counting.

statistics signature-hit enable

By default, IPS signature hit counting is disabled.

Configuring IPS whitelist

About this task

If false alarms exist in IPS logs, you can enable the IPS whitelist feature, and add the detected IPS signature IDs, URLs, or source IP addresses to the IPS whitelist. The IPS signature IDs, URLs, and source IP addresses are recorded in the IPS logs. The device permits packets matching the IPS signatures, URLs, or source IP addresses on the IPS whitelist to pass through, reducing false alarms.

If an IPS whitelist entry contains a signature ID, URL, and source IP address, or two of them, a packet matches this entry only when it matches all configured criteria.

Procedure

1. Enter system view.

   `system-view`

2. Enable the IPS whitelist feature.

   `ips whitelist enable`

   By default, the IPS whitelist feature is disabled.

3. Create an IPS whitelist entry and enter its view.

   `ips whitelist` *`entry-id`*

4. Configure the description for the IPS whitelist entry.

   `description` *`text`*

   By default, an IPS whitelist entry does not have any description.

5. Configure the IPS whitelist entry. Choose at least one of the following options to configure:
   - Add a signature ID to the IPS whitelist entry.

     `signature-id` *`sig-id`*

     By default, no signature ID exists in an IPS whitelist entry.
   - Add a URL to the IPS whitelist entry.

     `url match-type` { **`accurate`** | **`substring`** } *`url-text`*

     By default, no URL exists in an IPS whitelist entry.
   - Add a source IP address to the IPS whitelist entry.

     `source-address` { **`ip`** *`ipv4-address`* | **`ipv6`** *`ipv6-address`* }

     By default, no source IP address exists in an IPS whitelist entry.

6. Return to system view.

   `quit`

7. Activate the IPS whitelist configuration.

   `ips whitelist activate`

   After you create or edit an IPS whitelist entry that contains a URL, you must execute this command to have the configuration take effect.

Display and maintenance commands for IPS

Execute **`display`** commands in any view.

Task	Command
Display IPS policy information.	**`display ips policy`** *`policy-name`*

Display IPS signature library information.	**display ips signature library**
Display IPS signature information.	**display ips signature** [**pre-defined** | **user-defined** { **snort** | **user-config** }] [**direction** { **any** | **to-client** | **to-server** }] [**category** *category-name* | **fidelity** { **high** | **low** | **medium** } | **protocol** { **icmp** | **ip** | **tcp** | **udp** } | **severity** { **critical** | **high** | **low** | **medium** }] *
Display detailed information about an IPS signature.	**display ips signature** { **pre-defined** | **user-defined** } *signature-id*
Display information about IPS signatures that failed to be parsed during signature import.	**display ips signature user-defined parse-failed**

IPS configuration examples

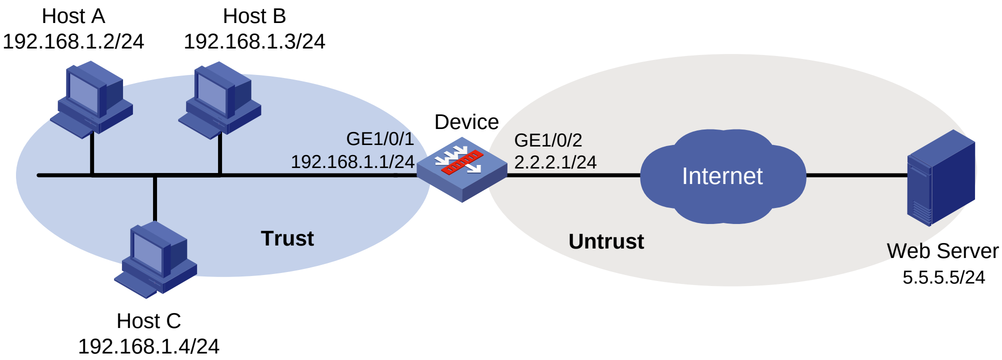
Example: Using the default IPS policy in a security policy

Network configuration

As shown in Figure 2, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

Configure the device to use the default IPS policy for attack detection and prevention.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Apply the default IPS policy to a DPI application profile and activate the IPS policy settings:

# Create a DPI application profile named **sec** and enter its view. Apply the default IPS policy to the DPI application profile and set the policy mode to **protect**.

```
[Device] app-profile sec
[Device-app-profile-sec] ips apply policy default mode protect
[Device-app-profile-sec] quit
```

# Activate the IPS policy settings.

```
[Device] inspect activate
```

5. Configure a security policy:

# Enter IPv4 security policy view. Create a rule named **trust-untrust** to permit the traffic from internal users to the external network and apply the IPS policy to the traffic between the internal users and the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Verify that the device can use the default IPS policy to detect and prevent known network attacks. (Details not shown.)

For example, if an incoming attack packet matches predefined IPS signature GNU_Bash_Local_Memory_Corruption_Vulnerability(CVE-2014-7187), the device automatically applies the signature actions (**reset** and **logging**) to the packet.

Example: Using a user-defined IPS policy in a security policy

Network configuration

As shown in Figure 3, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

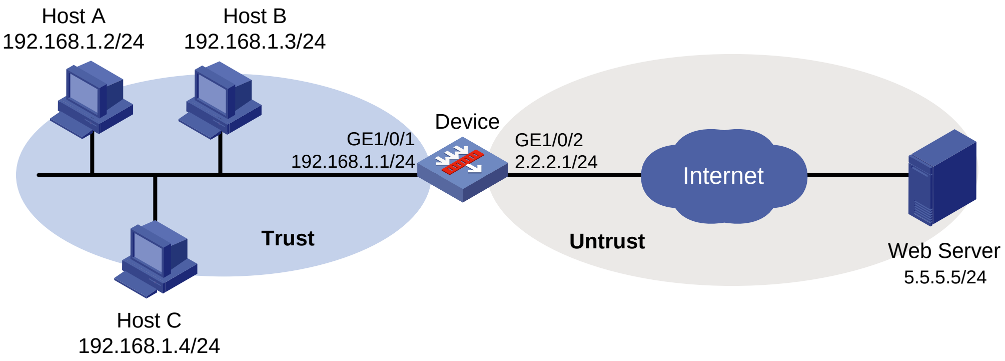
Perform the following tasks:

1. Create IPS policy **ips1** and modify its signature action and status settings as follows:
   - Enable predefined IPS signature 2 and specify actions **drop**, **capture**, and **logging** for the signature.

- Disable predefined IPS signature 4.
- Enable predefined IPS signature 6.

2. Apply IPS policy **ips1** to the zone pair between source security zone **Trust** and destination security zone **Untrust**.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure an IPS policy:

   # Create an IPS policy named **ips1** and enter its view. Configure the IPS policy to use all IPS signatures without discrimination of the target attribute. Enable predefined IPS signature 2 and specify actions **drop**, **capture**, and **logging** for the signature, disable predefined IPS signature 4, and enable predefined IPS signature 6.

   ```
   [Device] ips policy ips1
   [Device-ips-policy-ips1] protect-target all
   [Device-ips-policy-ips1] signature override pre-defined 2 enable drop capture logging
   [Device-ips-policy-ips1] signature override pre-defined 4 disable
   [Device-ips-policy-ips1] signature override pre-defined 6 enable
   [Device-ips-policy-ips1] quit
   ```

5. Apply IPS policy **ips1** to a DPI application profile, and activate the IPS policy settings:

# Create a DPI application profile named **sec**. Apply IPS policy **ips1** to the DPI application profile and set the policy mode to **protect**.

```
[Device] app-profile sec
[Device-app-profile-sec] ips apply policy ips1 mode protect
[Device-app-profile-sec] quit
```

# Activate the IPS policy settings.

```
[Device] inspect activate
```

6. Configure a security policy:

# Enter IPv4 security policy view. Create a rule named **trust-untrust** to permit the traffic from internal users to the external network and apply the IPS policy to the traffic between the internal users and the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Verify that IPS policy **ips1** is configured correctly.

```
<Device> display ips policy ips1
```

Example: Manually updating the IPS signature library

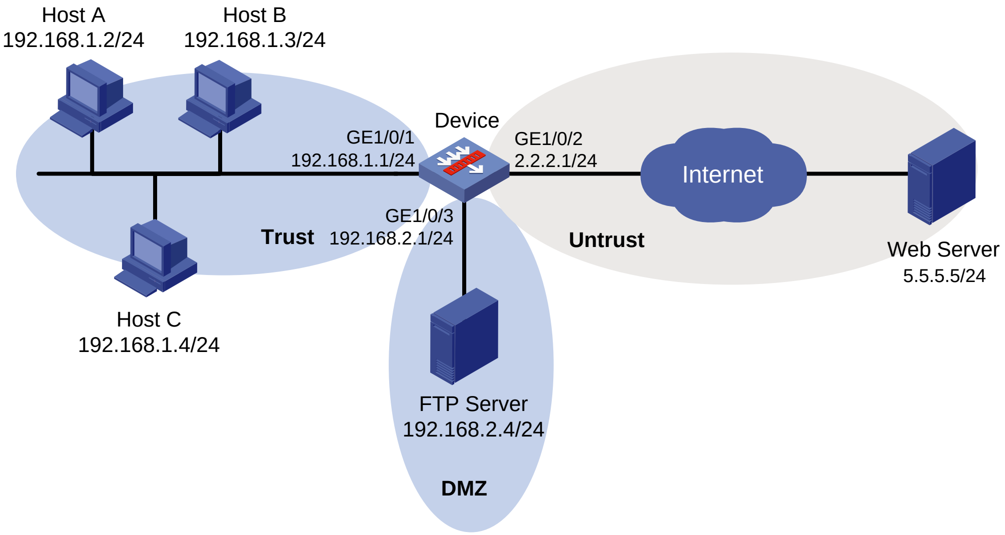
Network configuration

As shown in Figure 4, LAN users in security zone **Trust** can access the following resources:

- Internet resources in security zone **Untrust**.
- The FTP server at 192.168.2.4/24 in security zone **DMZ**. The FTP login username and password are **ips** and **123**, respectively.

Manually update the IPS signature library by using the latest IPS signature file stored on the FTP server.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
   [Device-security-zone-DMZ] quit
   ```

4. Configure a security policy:

   # Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **Untrust** so internal users can access external resources.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   ```

```
[Device-security-policy-ip-10-trust-untrust] quit
```

# Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **DMZ** so internal users can access the FTP server in the **DMZ** security zone.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-11-trust-dmz] source-zone trust
[Device-security-policy-ip-11-trust-dmz] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-11-trust-dmz] destination-zone dmz
[Device-security-policy-ip-11-trust-dmz] action pass
[Device-security-policy-ip-11-trust-dmz] quit
```

# Configure a security policy rule to permit the traffic between the device and the FTP server so the device can access the FTP server to obtain the signature file.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name ftplocalout
[Device-security-policy-ip-12-ftplocalout] source-zone local
[Device-security-policy-ip-12-ftplocalout] destination-zone dmz
[Device-security-policy-ip-12-ftplocalout] destination-ip-subnet 192.168.2.0 24
[Device-security-policy-ip-12-ftplocalout] application ftp
[Device-security-policy-ip-12-ftplocalout] application ftp-data
[Device-security-policy-ip-12-ftplocalout] action pass
[Device-security-policy-ip-12-ftplocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

5. Update the IPS signature library on the device by using IPS signature file **ips-1.0.8-encrypt.dat** on the FTP server.

```
[Device] ips signature update ftp://ips:123@192.168.2.4/ips-1.0.8-encrypt.dat
```

Verifying the configuration

# Verify that the device IPS signature library is updated.

```
<Device> display ips signature library
```

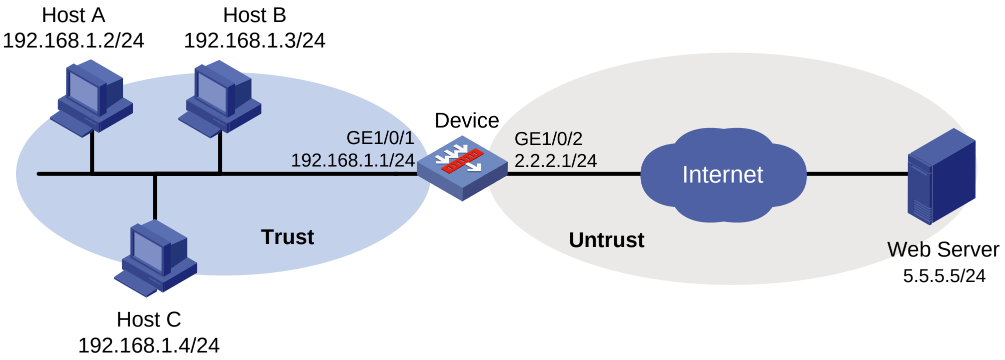
Example: Configuring automatic IPS signature library update

Network configuration

As shown in Figure 5, LAN users in security zone **Trust** can access Internet resources in security zone **Untrust**.

Configure the device to start automatically updating the local IPS signature library at a random time between 08:30 a.m. and 09:30 a.m. every Saturday.

Figure 5 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure DNS for the device to resolve the domain name of the official website into the correct IP address.

```
[Device] dns server 10.72.66.36
```

5. Configure a security policy:

   # Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **Untrust** so internal users can access external resources.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] quit
```

   # # Configure a security policy rule to permit the traffic from security zone **Local** to security zone **Untrust** so the device can access the official website to obtain the signature file.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name downloadlocalout
```

```
[Device-security-policy-ip-11-downloadlocalout] source-zone local
[Device-security-policy-ip-11-downloadlocalout] destination-zone untrust
[Device-security-policy-ip-11-downloadlocalout] action pass
[Device-security-policy-ip-11-downloadlocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

6. Configure automatic IPS signature library update:

# Enable automatic IPS signature library update, and configure the device to perform automatic update at a random time between 08:30 a.m. and 09:30 a.m. every Saturday.

```
[Device] ips signature auto-update
[Device-ips-autoupdate] update schedule weekly sat start-time 9:00:00 tingle 60
[Device-ips-autoupdate] quit
```

Verifying the configuration

# Verify that the device IPS signature library is updated as scheduled.

```
<Device> display ips signature library
```

Contents

Configuring URL filtering

About URL filtering

URL filtering controls access to the Web resources by filtering the URLs that the users visit.

URL

A URL is a reference to a resource that specifies the location of the resource on a network and a mechanism for retrieving it. The syntax of a URL is protocol://host [:port]/path/[;parameters][?query]#fragment. Figure 1 shows an example URL.

Figure 1 URL syntax

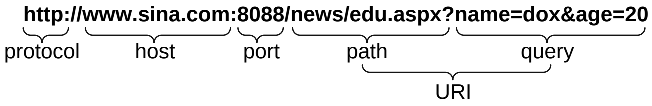


Table 1 describes the fields in a URL.

Table 1 URL field descriptions

Field	Description
protocol	Transmission protocol, such as HTTP.
host	Domain name or IP address of the server where the indicated resource is located.
[:port]	Optional field that identifies the port number of the transmission protocol. If this field is omitted, the default port number of the protocol is used.
/path/	String that identifies the directory or file where the indicated resource is stored. The path is a sequence of segments separated by zero or multiple forward slashes.
[parameters]	Optional field that contains special parameters.
[?query]	Optional field that contains parameters to be passed to the software for querying dynamic webpages. Each parameter is a <key>=<value> pair. Different parameters are separated by an ampersand (&).
URI	Uniform resource identifier that identifies a resource on a network.

URL filtering rule

A URL filtering rule matches URLs based on the content in the URI or hostname field.

URL filtering rule type

URL filtering provides the following types of URL filtering rules:

- **Predefined URL filtering rules**—Signature-based URL filtering rules. The device automatically generates them based on the local URL filtering signatures. In most cases, the predefined rules are sufficient for URL filtering.
- **User-defined URL filtering rules**—Regular expression- or text-based URL filtering rules that are manfully configured.

URL filtering rule matching method

A URL filtering rule supports the following URL matching methods:

- **Text-based matching**—Matches the hostname and URI fields of a URL against text patterns.

  When performing text-based matching for the hostname field of a URL, the device first determines if the text pattern contains the asterisk (*) wildcard character at the beginning or end.

  - If the text pattern does not contain the asterisk (*) wildcard character at the beginning or end, the hostname matching succeeds if the hostname of the URL matches the text pattern.
  - If the text pattern contains the asterisk (*) wildcard character at the beginning, the hostname matching succeeds if the hostname of the URL matches or ends with the text pattern without the wildcard character.
  - If the text pattern contains the asterisk (*) wildcard character at the end, the hostname matching succeeds if the hostname of the URL matches or starts with the text pattern without the wildcard character.
  - If the text pattern contains the asterisk (*) wildcard character at both the beginning and the end, the hostname matching succeeds if the hostname of the URL matches or includes the text pattern without the wildcard characters.

  Text-based matching for the URI field works in the same way that text-based matching for the hostname field works.

- **Regular expression-based matching**—Matches the hostname and URI fields of a URL against regular expressions. For example, if you set the regular expression for hostname matching to **sina.*cn**, URLs that carry the **news.sina.com.cn** hostname will be matched.

URL category

URL filtering provides the URL categorization feature to facilitate filtering rule management.

You can classify multiple URL filtering rules to a URL category and specify an action for the category. If a matching rule is in multiple URL categories, the system takes the action for the category with the highest severity level.

URL filtering supports the following types of URL categories:

- Predefined URL categories.

  The predefined URL categories contain the predefined URL filtering rules. Each predefined URL category has a unique severity level in the range of 1 to 999, and a category name that begins **Pre**-. Predefined URL categories cannot be modified.

  The device supports two levels of predefined URL categories: child URL category and parent URL category.

  A predefined parent URL category contains only predefined child URL categories.

- User-defined URL categories.

  You can manually create URL categories and configure filtering rules for them. The severity level of a user-defined URL category is in the range of 1000 to 65535. You can edit the filtering rules and change the severity level for a user-defined URL category.

URL filtering whitelist/blacklist rule

The device supports using URL-based whitelist and blacklist rules to filter packets. If the URL in a packet matches a blacklist rule, the packet is dropped. If the URL matches a whitelist rule, the packet is permitted to pass through.

URL filtering policy

A URL filtering policy can contain the following settings:

- URL categories and filtering actions. URL filtering actions include drop, permit, block source, reset, redirect, and logging.
- URL filtering whitelist and blacklist rules.
- URL filtering cloud query.

You can also specify the default action on packets that do not match any filtering rules (including URL categories, URL filtering whitelist and blacklist rules) in the policy.

URL filtering mechanism

URL filtering takes effect after you apply a URL filtering policy to a DPI application profile and use the DPI application profile in a security policy rule.

As shown in Figure 2, upon receiving a packet, the device performs the following operations:

1. The device compares the packet with the security policy rules.

   If the packet matches a rule that is associated with a URL filtering policy (through a DPI application profile), the device extracts the URL from the packet.

   For more information about security policies, see *Security Configuration Guide*.

2. The device compares the extracted URL with the whitelist and blacklist rules in the URL filtering policy.

   If both the whitelist and blacklist features are enabled, the device uses the following process to handle the packet:

   a. If the URL matches a whitelist rule, the packet is permitted to pass through.

   b. If the URL does not match a whitelist rule, the device identifies whether the URL matches a blacklist rule.

      – If the URL matches a blacklist rule, the packet is dropped.

      – If the URL does not match a blacklist rule, the device performs step 3.

   If only the whitelist feature is enabled, the device handles the packet as follows:

   ○ If the URL matches a whitelist rule, the packet is permitted to pass through.

   ○ If the URL does not match a whitelist rule, the device drops the packet.

   If both the whitelist and blacklist features are not enabled, the device performs step 3.

3. The device compares the extracted URL with the URL filtering rules in the URL filtering policy.

   a. If the URL matches a URL filtering rule that belongs to a user-defined URL category, the devices takes the action specified for the URL category. If the URL filtering rule belongs to multiple user-defined URL categories, the action specified for the URL category with the highest severity level apply.

      If no matching URL filtering rule belongs to a user-defined URL category, the device moves to step b.

   b. If the URL matches a URL filtering rule that belongs to a predefined URL category, the devices takes the action specified for the URL category.

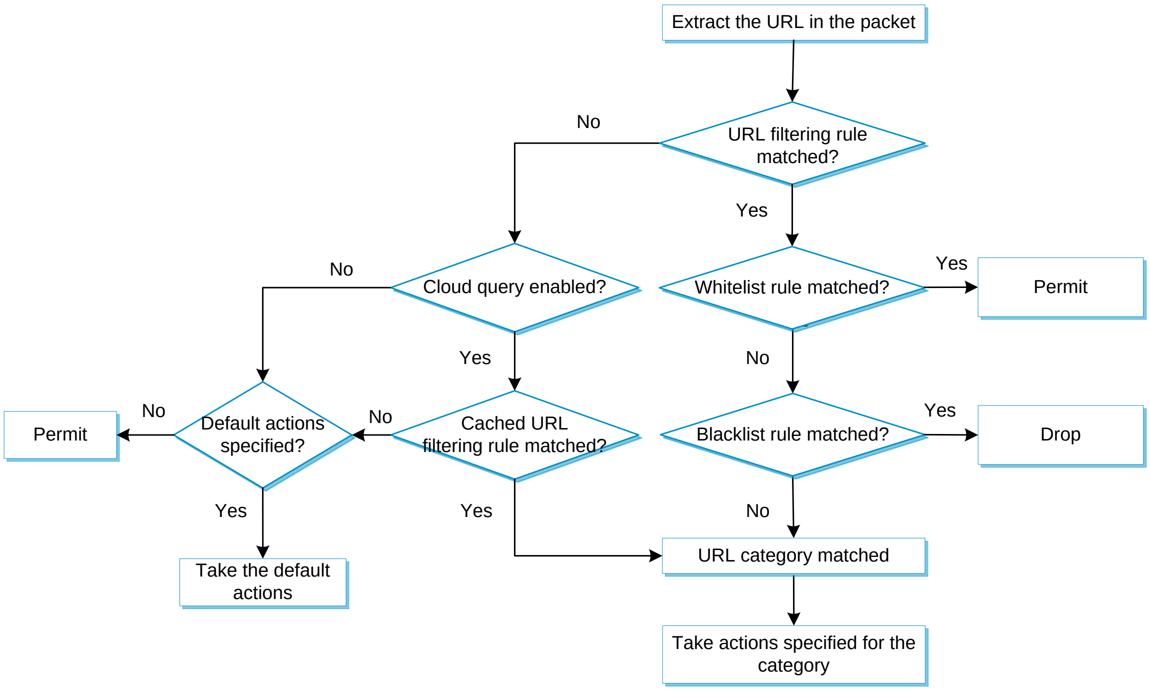
      If the URL filtering rule belongs to multiple predefined URL categories, the action specified for the URL category with the highest severity level apply.

4. If the URL does not match any rule in the policy, and cloud query is disabled in the URL filtering policy, the default action specified for the policy applies. If the default action is not configured, the device permits the packet to pass through.

   If the URL does not match any rule in the policy, and cloud query is enabled in the policy, the device handles the packet as follows:

- The device identifies whether the URL matches a cached URL filtering rule (history query result from the cloud server, including the URL and its category name).
  - If a matching cached rule is found for the URL, the device determines the action to take on the packet as described in step c of step 3.
  - If no matching cached rule is found for the URL, the default action specified for the policy applies. If the default action is not configured, the device permits the packet to pass through. In addition, the device sends the URL to the cloud server for further query and caches the query result.

Figure 2 URL filtering mechanism



URL filtering signature library management

The device uses the local URL filtering signature library to identify URLs in the HTTP packets.

You can update the device URL filtering signature library to the most up-to-date version or roll back the library to a version.

Updating the URL filtering signature library

The following methods are available for updating the URL filtering signature library on the device:

- Automatic update.

  The device periodically accesses the company's website and automatically downloads the most up-to-date URL filtering signature file to update its local signature library.

- Triggered update.

  The device downloads the most up-to-date URL filtering signature file from the company's website to update its local signature library immediately you trigger the operation.

- Manual update.

  Use this method when the device cannot connect to the company's website.

  You must manually download the most up-to-date URL filtering signature file from the company's website, and then use the file to update the signature library on the device.

Rolling back the URL filtering signature library

If filtering false alarms or filtering exceptions occur frequently, you can roll back the URL filtering signature library to the previous version or to the factory default version.

Restrictions: Licensing requirements for URL filtering

A license is required for URL filtering signature library update and URL filtering cloud query. If the license expires, the existing URL filtering signature library is still available but you cannot update the library on the device or perform a URL filtering cloud query task. For more information about licenses, see license management in *Fundamentals Configuration Guide*.

URL filtering tasks at a glance

To configure URL filtering:

1. (Optional.) Configuring a URL category
2. (Optional.) Configuring URL filtering cloud query
3. Configuring a URL filtering policy
4. (Optional.) Copying a URL filtering policy or category
5. Applying a URL filtering policy to a DPI application profile
6. (Optional.) Activating URL filtering policy and rule settings
7. Applying a DPI application profile to a security policy rule
8. Managing the URL filtering signature library
9. (Optional.) Managing the URL filtering signature library
10. (Optional.) Enabling DPI engine logging
11. (Optional.) Configuring URL filtering logging for resource access

Configuring a URL category

About this task

Perform this task to create a user-defined URL category and configure filtering rules for it to meet specific URL filtering requirements.

Restrictions and guidelines

When creating a URL category, you must assign a unique severity level in the range of 1000 to 65535 to the URL category. The larger the value, the higher the severity level.

Procedure

1. Enter system view.

   system-view

2. Create a URL category and enter its view.

   url-filter category *category-name* [**severity** *severity-level*]

   By default, the device provides predefined URL categories with names starting with **Pre-**.

   The name of a user-defined URL category cannot start with **Pre-**.

3. (Optional.) Configure a description for the URL category.

   description *text*

4. Configure URL filtering rules for the URL category. Choose the options to configure as needed:
   - Configure a URL filtering rule.

     rule *rule-id* **host** { **regex** *regex* | **text** *string* } [**uri** { **regex** *regex* | **text** *string* }]
   - (Optional.) Add the URL filtering rules of a predefined URL category to the URL category.

     include pre-defined *category-name*

     By default, a user-defined URL category does not contain the URL filtering rules of any predefined URL category.
5. (Optional.) Rename the URL category.

   rename *new-name*

Configuring URL filtering cloud query

About this task

The URL filtering cloud query feature enables the system to send URLs that do not match any local URL filtering rules to the cloud server for further query. This helps improves URL filtering accuracy for HTTP traffic.

The device caches the URL filtering rules returned from the cloud query server in the URL filtering cache. You can set the maximum number of rules that can be cached, and the minimum cache period for the cached rules. For more information about the cloud query server, see "Configuring the DPI engine."

Procedure

1. Enter system view.

   system-view
2. Specify the cloud query server.

   inspect cloud-server *host-name*
3. (Optional.) Set URL filtering cache size.

   url-filter cache size *cache-size*

   The URL filtering cache can cache a maximum of 16384 entries.
4. (Optional.) Set the minimum cache period for URL filtering rules.

   url-filter cache-time *value*

   By default, the minimum cache period is 10 seconds.
5. Enter the view of the URL filtering policy in which you want to enable cloud query.

   url-filter policy *policy-name*
6. Enable cloud query.

   cloud-query enable

   By default, cloud query is disabled in a URL filtering policy.

Configuring a URL filtering policy

About configuring a URL filtering policy

The URL filtering is implemented by URL filtering polices.

To configure a URL filtering policy, perform either of the following tasks:

- Configuring a category-based URL filtering policy

A category-based URL filtering policy contains the following settings:

- URL category-to-action mappings.
- Default action.
- (Optional.) Whitelist and blacklist rules.

- Configuring a whitelist-based URL filtering policy

Configuring a category-based URL filtering policy

Restrictions and guidelines

The **logging** keyword enables the URL filtering module to log URL filtering events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output URL filtering logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view URL filtering logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a URL filtering policy and enter its view.

   url-filter policy *policy-name*

3. Specify the actions for a URL category.

   category *category-name* **action** { **block-source** [**parameter-profile** *parameter-name*] | **drop** | **permit** | **redirect parameter-profile** *parameter-name* | **reset** } [**logging** [**parameter-profile** *parameter-name*]]

   By default, no actions are specified for a URL category.

   If a packet matches a rule that is in multiple URL categories, the system uses the actions for the category with the highest severity level.

4. (Optional.) Specify the default action on packets that do not match any rule in the policy.

   default-action { **block-source** [**parameter-profile** *parameter-name*] | **drop** | **permit** | **redirect parameter-profile** *parameter-name* | **reset** } [**logging** [**parameter-profile** *parameter-name*]]

5. (Optional.) Configure a whitelist or blacklist rule in the policy.

   add { **blacklist** | **whitelist** } [*id*] **host** { **regex** *host-regex* | **text** *host-name* } [**uri** { **regex** *uri-regex* | **text** *uri-name* }]

6. (Optional.) Enable the referer whitelist.

   referer-whitelist enable

   By default, the referer whitelist is enabled. It allows an HTTP or HTTPS request to pass through if its referer header matches a whitelist rule.

7. (Optional.) Rename the URL filtering policy.

   rename *new-name*

Configuring a whitelist-based URL filtering policy

About the task

This feature allows only the HTTP or HTTPS requests that match the whitelist rules to pass through. When you do not want to perform any other configurations, such as URL categories, URL filtering actions, and URL filtering policy default action, you can use this feature.

With this feature enabled, the device allows users to access only the Web resources added to the whitelist rules, and other Web resources are not allowed to access.

Procedure

1. Enter system view.

   `system-view`

2. Create a URL filtering policy and enter its view.

   `url-filter policy` *`policy-name`*

3. Configure a whitelist rule in the policy.

   `add whitelist` [*`id`*] **`host`** { **`regex`** *`host-regex`* | **`text`** *`host-name`* } [**`uri`** { **`regex`** *`uri-regex`* | **`text`** *`uri-name`* }]

4. (Optional.) Enable the referer whitelist.

   `referer-whitelist enable`

   By default, the referer whitelist is enabled. It allows an HTTP or HTTPS request to pass through if its referer header matches a whitelist rule.

5. Enable URL whitelist-only filtering.

   `whitelist-only enable`

   By default, URL whitelist-only filtering is disabled.

Copying a URL filtering policy or category

Copying a URL filtering policy

About this task

You can create a new URL filtering policy by copying an existing one.

Procedure

1. Enter system view.

   `system-view`

2. Create a URL filtering policy and enter its view.

   `url-filter copy policy` *`old-name new-name`*

Copying a URL filtering category

About this task

You can create a new URL category by copying an existing one.

Restrictions and guidelines

When you copy a URL category, be sure to assign a unique severity level to the new URL category.

Procedure

1. Enter system view.

   `system-view`

2. Copy a URL category.

   `url-filter copy category` *`old-name new-name`* **`severity`** *`severity-level`*

Applying a URL filtering policy to a DPI application profile

About this task

A URL filtering policy must be applied to a DPI application profile to take effect.

Restrictions and guidelines

A DPI application profile can use only one URL filtering policy. If you apply different URL filtering policies to the same DPI application profile, only the most recent configuration takes effect.

Procedure

1. Enter system view.

   `system-view`

2. Enter DPI application profile view.

   `app-profile` *`app-profile-name`*

   For more information about this command, see DPI engine commands in *DPI Command Reference*.

3. Assign a URL filtering policy to the DPI application profile.

   `url-filter apply policy` *`policy-name`*

   By default, no URL filtering policy is applied to the DPI application profile.

Activating URL filtering policy and rule settings

About this task

By default, the system will detect whether another configuration change (such as creation, modification, or deletion) occurs within a 20-second interval after a change to the URL filtering policy and rule settings:

- If no configuration change occurs within the interval, the system will perform an activation operation at the end of the next 20-second interval to make the configuration take effect.
- If a configuration change occurs within the interval, the system continues to detect configuration changes at 20-second intervals.

To immediately activate a configuration change, execute the **`inspect activate`** command.

For more information about activating DPI service module configuration, see "Configuring the DPI engine."

Procedure

1. Enter system view.

   `system-view`

2. Activate URL filtering policy and rule settings.

   `inspect activate`

By default, the system automatically activates changed URL filtering policy and rule settings for them to take effect.

△ **CAUTION:**

This command can cause temporary outage for DPI services. Services based on the DPI services might also be interrupted. For example, security policies cannot control application access.

Applying a DPI application profile to a security policy rule

1. Enter system view.
   system-view
2. Enter security policy view.
   security-policy { **ip** | **ipv6** }
3. Enter security policy rule view.
   rule { *rule-id* | [*rule-id*] **name** *rule-name* }
4. Set the rule action to pass.
   action pass

   The default rule action is **drop**.
5. Use a DPI application profile in the rule.
   profile *app-profile-name*

   By default, no DPI application profile is used in a security policy rule.

Managing the URL filtering signature library

You can update or roll back the version of the URL filtering signature library on the device.

Restrictions and guidelines

- Do not delete the **/dpi/** folder in the root directory of the storage medium.
- Do not perform URL filtering signature update and rollback when the device's free memory is below the normal state threshold. For more information about device memory thresholds, see device management in *Fundamentals Configuration Guide*.
- For successful automatic and immediate signature update, make sure the device can resolve the domain name of the company's website into an IP address through DNS. For more information about DNS, see DNS configuration in *Layer 3—IP Services Configuration Guide.*
- Update only one signature library at a time. Do not perform signature library update until the existing signature library update is completed.

Scheduling automatic URL filtering signature library update

About this task

You can schedule automatic URL filtering signature library update if the device can access the signature database services on the company's website. The device periodically obtains the latest signature file from the company's website to update its local signature library as scheduled.

Procedure

1. Enter system view.

   system-view

2. Enable automatic URL filtering signature library update and enter automatic URL filtering signature library update configuration view.

   url-filter signature auto-update

   By default, automatic URL filtering signature library update is disabled.

3. Schedule the update time.

   update schedule { **daily** | **weekly** { **fri** | **mon** | **sat** | **sun** | **thu** | **tue** | **wed** } } **start-time** *time* **tingle** *minutes*

   By default, the device starts to update the URL filtering signature at a random time between 01:00:00 and 03:00:00 every day.

Triggering an immediate URL filtering signature update

About this task

Anytime you find a release of new signature version on the company's website, you can trigger the device to immediately update the local signature library.

Procedure

1. Enter system view.

   system-view

2. Trigger an automatic URL filtering signature library update.

   url-filter signature auto-update-now

Performing a URL filtering signature manual update

About this task

If the device cannot access the signature database services on the company's website, use one of the following methods to manually update the URL filtering signature library on the device:

- **Local update**—Updates the URL filtering signature library on the device by using the locally stored update URL filtering signature file.

  Store the update file on the master device for successful signature library update.

- **FTP/TFTP update**—Updates the URL filtering signature library on the device by using the file stored on the FTP or TFTP server.

Procedure

1. Enter system view.

   system-view

2. Manually update the URL filtering signature library on the device.

   url-filter signature update *file-path*

△ **CAUTION:**

Select a signature file according to the memory size and software version of the device. NSFOCUS provides signature files separately for high-memory (equal to or higher than 8 GB) and low-memory (lower than 8 GB) devices and for different software versions. If you use a signature file applicable to high-memory devices to update the URL filtering signature library on a low-memory device, exceptions might occur on the low-memory device. As a best practice, use a signature file that is compatible with the software version and memory size of the device to update the URL filtering signature library on the device.

Rolling back the URL filtering signature library

About this task

If a URL filtering signature library update causes exceptions or a high false alarm rate, you can roll back the URL filtering signature library.

Before rolling back the URL filtering signature library, the device backs up the current signature library as the "previous version." For example, the previous library version is V1 and the current library version is V2. If you perform a rollback to the previous version, library version V1 becomes the current version and library version V2 becomes the previous version. If you perform a rollback to the previous version again, the library rolls back to library version V2.

Procedure

1. Enter system view.

   `system-view`

2. Roll back the URL filtering signature library to the previous version or to the factory default version.

   `url-filter signature rollback { factory | last }`

Enabling DPI engine logging

About this task

You can enable DPI engine logging for audit purposes. Log messages generated by DPI engine are output to the device information center. The information center then sends the messages to designated destinations based on log output rules. For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable DPI engine logging.

   `url-filter log enable`

   By default, DPI engine logging is disabled.

Configuring URL filtering logging for resource access

About URL filtering logging for resource access

URL filtering logs user access to resources after you specify the logging action for a URL category or as a default action for a URL filtering policy.

You can use either of the following methods to configure URL filtering to log access to specific types of resources:

- Configure URL filtering to log access to only resources in the root directories of websites.
- Enable or disable URL filtering logging for access to resources of specific types.

Logging access to only resources in the root directories of websites

1. Enter system view.

   system-view

2. Configure URL filtering to log only access to resources in the root directories of websites.

   url-filter log directory root

   By default, URL filtering logs access to Web resources in all directories.

Disabling logging for access to resources of specific types

1. Enter system view.

   system-view

2. Disable URL filtering logging for access to resources of a specific resource type.

   - Disable logging for access to resources of a predefined resource type.

     url-filter log except pre-defined { **css** | **gif** | **ico** | **jpg** | **js** | **png** | **swf** | **xml** }

   - Disable logging for access to resources of a user-defined resource type.

     url-filter log except user-defined *text*

   By default, URL filtering logs access to all resources except for resources of the predefined resource types (including CSS, GIF, ICO, JPG, JS, PNG, SWF, and XML resources).

Enabling HTTPS URL filtering

About the task

By default, the device supports only the HTTP URL filtering. To enable filtering on HTTPS traffic, use either of the following methods:

- Use SSL decryption to decrypt the HTTPS traffic and then perform HTTP URL filtering on the decrypted traffic. For more information about SSL decryption, see proxy policy configuration in *DPI Configuration Guide*.

  SSL decryption involves a large number of encryption and decryption operations, which might downgrade device forwarding performance. As a best practice, use this method only when the device must perform URL filtering on HTTPS traffic.

- Enable HTTPS URL filtering. This feature performs URL filtering on undecrypted HTTPS traffic. The device directly detects the Client Hello message from the client, and extracts the server name from the Sever Name Indication (SNI) extension to match the URL filtering policy.

Restrictions and guidelines

If SSL decryption is configured, this feature does not take effect.

In HTTPS URL filtering, only the hostname match criterion in a URL filtering rule takes effect. The URI match criterion does not take effect.

This feature takes effect only when the hostname field in the URL is the server's domain name. This feature does not apply to the HTTPS traffic if the hostname field is an IP address.

This feature does not take effect in the following situations:

- The client browser enables TLS 1.3 downgrade enhancement mechanism, because the SNI extension will be encrypted.
- The HTTPS packets do not have the SNI extension.

Procedure

1. Enter system view.

   system-view

2. Create a URL filtering policy and enter its view.

   url-filter policy *policy-name*

3. Enable HTTPS URL filtering.

   https-filter enable

   By default, HTTPS URL filtering is disabled, and the device supports only the HTTP URL filtering.

Display and maintenance commands for URL filtering

Execute **display** commands except the **display url-reputation attack-category** command in any view and **reset** commands in user view.

Execute the **display url-reputation attack-category** command in URL filtering policy view.

Task	Command
Display URL filtering cache information.	**display url-filter cache**
Display URL category information.	**display url-filter** { **category** | **parent-category** } [**verbose**]
Display information about the URL filtering signature library.	**display url-filter signature library**
Display URL filtering statistics.	**display url-filter statistics**
Clear URL filtering statistics.	**reset url-filter statistics**

URL filtering configuration examples

Example: Using a URL filtering policy in a security policy

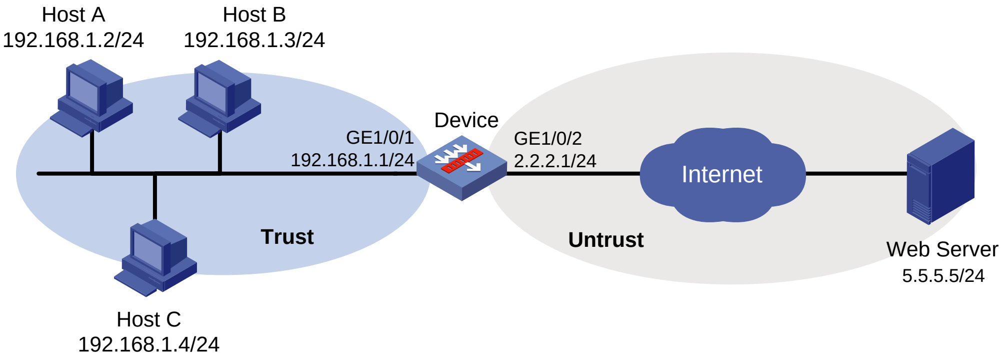
Network configuration

As shown in Figure 3, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

Configure a URL filtering policy on the device so the device performs the following operations:

- Permits LAN users in security zone **Trust** to access website **http://www.sina.com** on the Web server.
- Drops and logs packets that match the Pre-Game URL category.
- Drops and logs packets that do not match any filtering rule in the URL filtering policy.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure URL filtering:

   # Create user-defined URL category **news**, set its severity level to **2000**, and create URL filtering rule 1 to match HTTP packets that contain host name **www.sina.com** in the URL.

   ```
   [Device] url-filter category news severity 2000
   [Device-url-filter-category-news] rule 1 host text www.sina.com
   [Device-url-filter-category-news] quit
   ```

   # Create a URL filtering policy named **urlnews**. Specify action **permit** for URL category **news** and action **drop** for predefined URL category **Pre-Games**, enable logging for the matching packets, and set the default actions to **drop** and **logging**.

   ```
   [Device] url-filter policy urlnews
   [Device-url-filter-policy-urlnews] category news action permit
   [Device-url-filter-policy-urlnews] category Pre-Games action drop logging
   [Device-url-filter-policy-urlnews] default-action drop logging
   [Device-url-filter-policy-urlnews] quit
   ```

5. Apply URL filtering policy **urlnews** to a DPI application profile and activate the IPS policy settings:

# Create a DPI application profile named **sec**, and apply URL filtering policy **urlnews** to the DPI application profile.

```
[Device] app-profile sec
[Device-app-profile-sec] url-filter apply policy urlnews
[Device-app-profile-sec] quit
```

# Activate the URL filtering policy and rule settings.

```
[Device] inspect activate
```

6. Configure a security policy:

# Enter IPv4 security policy view. Create a rule named **trust-untrust** to permit the traffic from internal users to the external network and apply the URL filtering policy to the traffic between the internal users and the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Verify that LAN users in security zone **Trust** can access website **http://www.sina.com** on the Web server. (Details not shown.)

# Verify that the device drops and logs LAN users' HTTP requests to game resources. (Details not shown.)

Example: Manually updating the URL filtering signature library

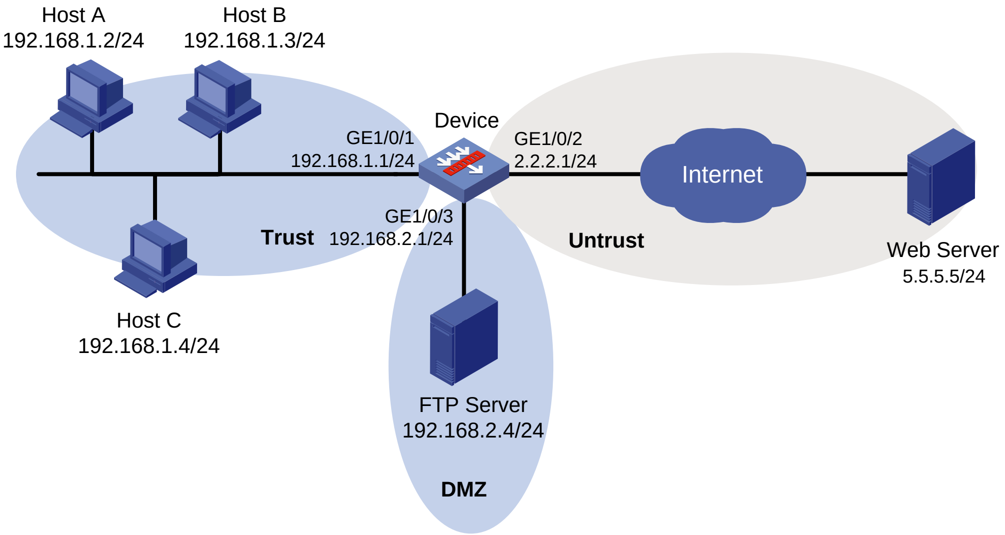
Network configuration

As shown in Figure 4, LAN users in security zone **Trust** can access the following resources:

- Internet resources in security zone **Untrust**.
- The FTP server at 192.168.2.4/24 in security zone **DMZ**. The FTP login username and password are **url** and **123**, respectively.

Manually update the URL filtering signature library on the device by using the latest URL filtering signature file (**url-1.0.2-encrypt.dat**) stored on the FTP server.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
   [Device-security-zone-DMZ] quit
   ```

4. Configure a security policy:

   # Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **Untrust** for the internal users to access external resources.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   ```

```
[Device-security-policy-ip-10-trust-untrust] quit
```

# Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **DMZ** for the internal users to access the FTP server in the **DMZ** security zone.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-dmz
[Device-security-policy-ip-11-trust-dmz] source-zone trust
[Device-security-policy-ip-11-trust-dmz] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-11-trust-dmz] destination-zone dmz
[Device-security-policy-ip-11-trust-dmz] action pass
[Device-security-policy-ip-11-trust-dmz] quit
```

# Configure a security policy rule to permit the traffic between the FTP server and the device so the device can access the FTP server to obtain the signature file.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name downloadlocalout
[Device-security-policy-ip-12-downloadlocalout] source-zone local
[Device-security-policy-ip-12-downloadlocalout] destination-zone dmz
[Device-security-policy-ip-12-downloadlocalout] destination-ip-subnet 192.168.2.0
24
[Device-security-policy-ip-12-downloadlocalout] application ftp
[Device-security-policy-ip-12-downloadlocalout] application ftp-data
[Device-security-policy-ip-12-downloadlocalout] action pass
[Device-security-policy-ip-12-downloadlocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

5. Update the URL filtering library on the device by using URL filtering signature file **url-1.0.2-encrypt.dat** on the FTP server.

```
[Device] url-filter signature update
ftp://url:123@192.168.2.4/url-1.0.2-encrypt.dat
```

Verifying the configuration

# Verify that the URL filtering signature library on the device is updated successfully.

```
<Device> display url-filter signature library
```

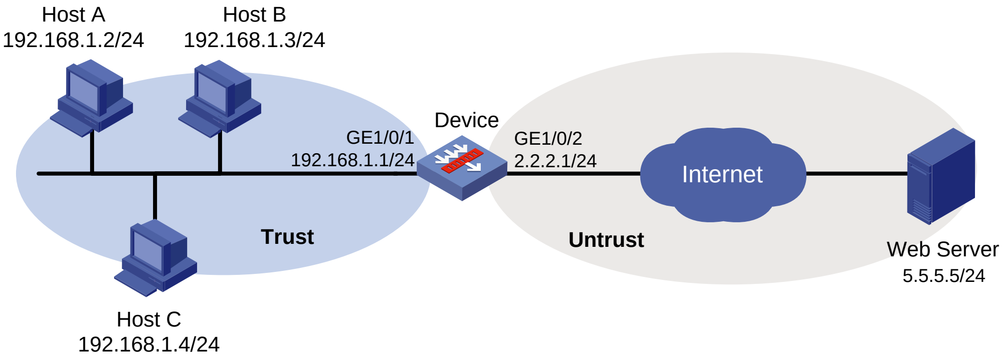
Example: Configuring automatic URL filtering signature library update

Network configuration

As shown in Figure 5, LAN users in security zone **Trust** can access Internet resources in security zone **Untrust**.

Configure the device to start automatically updating the local URL filtering signature library at a random time between 08:30 a.m. and 09:30 a.m. every Saturday.

Figure 5 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure DNS for the device to resolve the domain name of the official website into the correct IP address.

   ```
   [Device] dns server 10.72.66.36
   ```

5. Configure a security policy:

   # Configure a security policy rule to permit the traffic from security zone **Trust** to security zone **Untrust** for the internal users to access external resources.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   [Device-security-policy-ip-10-trust-untrust] quit
   ```

   # Configure a security policy rule to permit the traffic from security zone **Local** to security zone **Untrust** so the device can access the official website to obtain the signature file.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name downloadlocalout
   ```

```
[Device-security-policy-ip-11-downloadlocalout] source-zone local
[Device-security-policy-ip-11-downloadlocalout] destination-zone untrust
[Device-security-policy-ip-11-downloadlocalout] action pass
[Device-security-policy-ip-11-downloadlocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

6. Configure automatic URL filtering signature library update:

# Enable automatic URL filtering signature library update. Configure the device to perform automatic update at a random time between 08:30 a.m. and 09:30 a.m. every Saturday.

```
[Device] url-filter signature auto-update
[Device-url-filter-autoupdate] update schedule weekly sat start-time 9:00:00 tingle 
60
[Device-url-filter-autoupdate] quit
```

Verifying the configuration

# Verify that the device URL filtering signature library is updated as scheduled.

```
<Device> display url-filter signature library
```

Contents

Configuring data filtering

About data filtering

Data filtering filters packets based on application layer information. You can use data filtering to effectively prevent leakage of internal information, distribution of illegal information, and unauthorized access to the Internet.

Data filtering supports filtering packets of the following protocols:

- HTTP.
- FTP.
- SMTP.
- IMAP.
- NFS.
- POP3.
- RTMP.
- SMB.

Basic concepts

Keyword match pattern

The device provides predefined keyword match patterns and allows you to create user-defined keyword match patterns in a keyword group.

- **Predefined pattern**—Includes the phone number, bank card number, credit card number, and ID card number patterns. These patterns can be used to identify packets that contain phone numbers, bank card numbers, credit card numbers, and ID card numbers.
- **User-defined pattern**—A text- or regular expression-based string to identify patterns in the application layer data of packets.

Keyword group

A keyword group is a group of keyword match patterns.

Data filtering rule

A data filtering rule contains a set of filtering criteria for matching packets, including keyword group, traffic direction, and application layer protocol. You can specify the actions to take on packets matching a data filtering rule. Supported actions include drop, permit, and logging. A packet must match all the filtering criteria for the actions specified for the rule to apply.

Data filtering mechanism

Data filtering takes effect after you apply a data filtering policy to a DPI application profile and use the DPI application profile in a security policy rule.

1. Compares the packet with the security policy rules.

If the packet matches a rule that is associated with a data filtering policy (through a DPI application profile), the device extracts the application layer information from the packet.

For more information about security policies, see *Security Configuration Guide*.

2. Determines the actions to take on the packet by comparing the extracted application layer information with the data filtering rules in the data filtering policy:

- If the packet does not match any data filtering rules in the policy, the device permits the packet to pass.
- If the packet matches only one rule, the device takes the actions specified for the rule.
- If the packet matches multiple rules, the device determines the actions as follows:
  - If the matching rules have both the permit and drop actions, the device takes the drop action.
  - If the logging action is specified for any of the matching rules, the device logs the packet.

Data filtering tasks at a glance

To configure data filtering, perform the following tasks:

3. Configuring a keyword group
4. Configuring a data filtering policy
5. Applying a data filtering policy to a DPI application profile
6. (Optional.) Activating data filtering policy and rule settings
7. Applying a DPI application profile to a security policy rule

Configuring a keyword group

About this task

A keyword group is a group of keyword match patterns. A keyword match pattern is a text or regular expression string that matches packets based on application layer data.

A packet matches a keyword group if it matches any keyword match pattern in the group.

Procedure

1. Enter system view.

   `system-view`

2. Create a keyword group and enter its view.

   `data-filter keyword-group` *`keywordgroup-name`*

3. (Optional.) Configure a description for the keyword group.

   `description` *`string`*

   By default, a keyword group does not have a description.

4. Configure keyword match patterns:
   - Create a user-defined keyword match pattern.

     `pattern` *`pattern-name`* { **`regex`** | **`text`** } *`pattern-string`*

     By default, a keyword group does not contain any user-defined keyword match patterns.
   - Enable a predefined keyword match pattern.

     `pre-defined-pattern name { bank-card-number | credit-card-number | id-card-number | phone-number }`

     By default, no predefined patterns are enabled in a keyword group.

Configuring a data filtering policy

About this task

A data filtering policy can contain a maximum of 32 data filtering rules. Each rule defines a set of filtering criteria and actions for matching packets. The filtering criteria include:

- One keyword group.
- One or more application layer protocols.
- Traffic direction.

Restrictions and guidelines

Data filtering rules applied to the NFS protocol take effect only on NFSv3 traffic.

Data filtering rules applied to the SMB protocol take effect only on SMBv1 and SMBv2 traffic.

The **logging** keyword enables the data filtering module to log packet matching events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output data filtering logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view data filtering logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a data filtering policy and enter its view.

   data-filter policy *policy-name*

3. (Optional.) Configure a description for the data filtering policy.

   description *string*

   By default, a data filtering policy does not have a description.

4. Create a data filtering rule and enter its view.

   rule *rule-name*

5. Specify a keyword group for the data filtering rule.

   keyword-group *keywordgroup-name*

   By default, a data filtering rule does not contain any keyword group.

6. Specify the application layer protocols to which the data filtering rule applies.

   application { **all** | **type** { **ftp** | **http** | **imap** | **nfs** | **pop3** | **rtmp** | **smb** | **smtp** } * }

   By default, no applicable application layer protocols are specified for a data filtering rule.

7. Specify the traffic directions to which the data filtering rule applies.

   direction { **both** | **download** | **upload** }

   By default, a data filtering rule applies to upload traffic.

8. Specify the actions to take on matching packets.

```
action { drop | permit } [ logging ]
```

The default action of a data filtering rule is **drop**.

Applying a data filtering policy to a DPI application profile

About this task

A data filtering policy must be applied to a DPI application profile to take effect.

A DPI application profile can use only one data filtering policy. If you apply different data filtering policies to the same DPI application profile, only the most recent configuration takes effect.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DPI application profile view.

   ```
   app-profile profile-name
   ```

   For more information about this command, see DPI engine commands in *DPI Command Reference*.

3. Apply a data filtering policy to the DPI application profile.

   ```
   data-filter apply policy policy-name
   ```

   By default, no data filtering policy is applied to the DPI application profile.

Activating data filtering policy and rule settings

About this task

By default, the system will detect whether another configuration change (such as creation, modification, or deletion) occurs within a 20-second interval after a change to the data filtering policy and rule settings:

- If no configuration change occurs within the interval, the system performs an activation operation at the end of the next 20-second interval to make the configuration take effect.
- If a configuration change occurs within the interval, the system continues to periodically detect whether configuration changes occur within next 20-second intervals.

To activate the policy and rule configurations immediately, you can execute the **inspect activate** command.

For more information about configuration activation for DPI service modules, see "Configuring DPI engine."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Activate data filtering policy and rule settings.

   ```
   inspect activate
   ```

   By default, data filtering policy and rule settings will be activated automatically.

CAUTION:

This command can cause temporary outage for DPI services. Services based on the DPI services might also be interrupted. For example, security policies cannot control application access.

Applying a DPI application profile to a security policy rule

3. Enter system view.

   system-view

4. Enter security policy view.

   security-policy { **ip** | **ipv6** }

5. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

6. Set the rule action to pass.

   action pass

   The default rule action is **drop**.

7. Use a DPI application profile in the rule.

   profile *app-profile-name*

   By default, no DPI application profile is used in a security policy rule.

Data filtering configuration examples

Example: Using a data filtering policy in a security policy

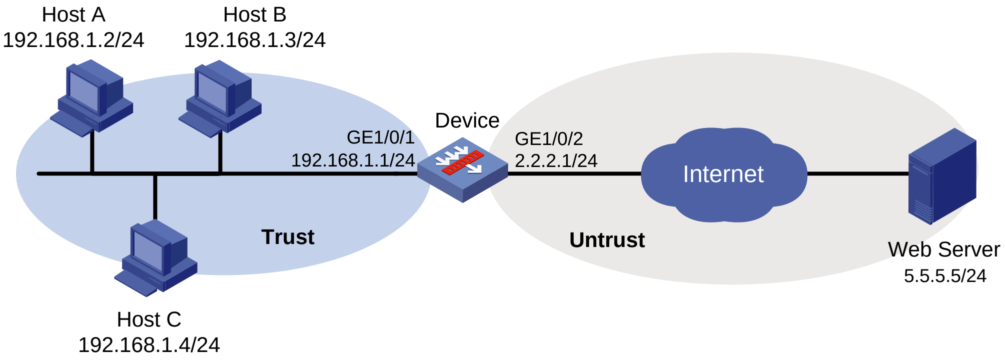
Network configuration

As shown in Figure 1, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

Configure data filtering on the device so the device performs the following operations:

- Blocks HTTP packets that contain the **uri** or **abc.*abc** string in the URI field or message body.
- Blocks download FTP traffic that contains the **http://www.abcd.com/** string.
- Logs the blocked packets.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure data filtering:

   a. Configure keyword groups:

      # Create a keyword group named **kg1** and create two keyword match patterns that match the **uri** text string and the **abc.*abc** regular expression string, respectively.

      ```
      [Device] data-filter keyword-group kg1
      [Device-data-filter-kgroup-kg1] pattern 1 text uri
      [Device-data-filter-kgroup-kg1] pattern 2 regex abc.*abc
      [Device-data-filter-kgroup-kg1] quit
      ```

      # Create a keyword group named **kg2** and create a keyword match pattern that matches the **http://www.abcd.com/** text string.

      ```
      [Device] data-filter keyword-group kg2
      [Device-data-filter-kgroup-kg2] pattern 1 text www.abcd.com
      [Device-data-filter-kgroup-kg2] quit
      ```

   b. Configure a data filtering policy:

      # Create a data filtering rule named **r1** and configure it to drop and log both upload and download HTTP traffic that matches keyword group **kg1**.

```
[Device] data-filter policy p1
[Device-data-filter-policy-p1] rule r1
[Device-data-filter-policy-p1-rule-r1] keyword-group kg1
[Device-data-filter-policy-p1-rule-r1] application type http
[Device-data-filter-policy-p1-rule-r1] direction both
[Device-data-filter-policy-p1-rule-r1] action drop logging
[Device-data-filter-policy-p1-rule-r1] quit
```

# Create a data filtering rule named **r2** and configure it to drop and log download FTP traffic that matches keyword group **kg2**.

```
[Device-data-filter-policy-p1] rule r2
[Device-data-filter-policy-p1-rule-r2] keyword-group kg2
[Device-data-filter-policy-p1-rule-r2] application type ftp
[Device-data-filter-policy-p1-rule-r2] direction download
[Device-data-filter-policy-p1-rule-r2] action drop logging
[Device-data-filter-policy-p1-rule-r2] quit
```

5. Configure a DPI application profile and activate the data filtering policy and rule settings:

   # Create a DPI application profile named **sec** and apply data filtering policy **p1** to the DPI application profile.

   ```
   [Device] app-profile sec
   [Device-app-profile-profile1] data-filter apply policy p1
   [Device-app-profile-profile1] quit
   ```

   # Activate the data filtering policy and rule settings.

   ```
   [Device] inspect activate
   ```

6. Configure a security policy:

   # Create a security policy rule named **trust-untrust.** Configure the rule to apply DPI application profile **sec** to packets from security zone **Trust** to security zone **Untrust** with source subnet address **192.168.1.0/24**.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   [Device-security-policy-ip-10-trust-untrust] profile sec
   [Device-security-policy-ip-10-trust-untrust] quit
   ```

   # Activate rule matching acceleration.

   ```
   [Device-security-policy-ip] accelerate enhanced enable
   [Device-security-policy-ip] quit
   ```

Verifying the configuration

# Verify that the device blocks and logs HTTP packets and FTP packets that meet the specified criteria. (Details not shown.)

Contents

Configuring file filtering

About file filtering

The file filtering feature filters files based on file extensions. You can configure file filtering to perform actions on files based on the file extensions.

File filtering supports filtering packets of the following protocols:

- HTTP.
- FTP.
- SMTP.
- IMAP.
- NFS.
- POP3.
- RTMP.
- SMB.

Basic concepts

File type match pattern

A file type match pattern identifies a type of files by file extension.

File type group

A file type group can contain a maximum of 32 file type match patterns. A file matches a file type group if it matches a pattern in the group.

File filtering rule

A file filtering rule contains a set of filtering criteria for matching files, including file type group, traffic direction, and application layer protocol. You can specify the actions to take on packets matching a file filtering rule. Supported actions include drop, permit, and logging. A file must match all the filtering criteria for the actions specified for the rule to apply.

File filtering mechanism

File filtering takes effect after you apply a file filtering policy to a DPI application profile and use the DPI application profile in a security policy rule.

Upon receiving a packet of a protocol that file filtering supports, the device performs the following operations:

1. Compares the packet with the security policy rules.

   If the packet matches a rule that is associated with a file filtering policy (through a DPI application profile), the device submits the packet to the DPI engine for file filtering processing.

   For more information about security policies, see *Security Configuration Guide*.
2. Extracts and records the file extension in the packet.
3. Identifies the real file extension and compares it with the recorded file extension:
   - If the two file extensions match or if the real file extension cannot be identified, the device proceeds to step 4.

- If the two file extensions do not match, the device checks the action specified for packets with files carrying false extensions.
  - If the **Drop** action is specified, the device drops the packet directly.
  - If the **Permit** action is specified, the device proceeds to step 4 to perform file filtering inspection based on the real file extension.

4. Determines the actions to take on the packet by comparing the packet attributes (file extension, application layer application, and traffic direction) with the file filtering rules in the file filtering policy:
   - If the packet does not match any file filtering rules in the policy, the device permits the packet to pass.
   - If the packet matches only one rule, the device takes the actions specified for the rule.
   - If the packet matches multiple rules, the device determines the actions as follows:
     - If the matching rules have both the **permit** and **drop** actions, the device takes the **drop** action.
     - The **logging** action is taken if it is specified for any of the matching rules.

File filtering tasks at a glance

To configure data filtering, perform the following tasks:

1. Configuring a file type group
2. Configuring a file filtering policy
3. Setting the action for packets with files carrying false extensions
4. Applying a file filtering policy to a DPI application profile
5. (Optional.) Activating file filtering policy and rule settings
6. Applying a DPI application profile to a security policy rule

Configuring a file type group

About this task

A file type group is a group of file type match patterns. A file type match pattern is a text or regular expression string that matches files by file extension.

A file matches a file type group if it matches a pattern in the group.

Procedure

1. Enter system view.
   system-view
2. Create a file type group and enter its view.
   file-filter policy *policy-name*
3. (Optional.) Configure a description for the file type group.
   description *string*
   By default, a file type group does not have a description.
4. Configure a file type match pattern.
   pattern *pattern-name* **text** *pattern-string*
   By default, a file type group does not contain any file type match patterns.

Configuring a file filtering policy

About this task

A file filtering policy can contain a maximum of 32 file filtering rules. Each rule defines a set of filtering criteria and the actions for matching packets. The filtering criteria include:

- One file type group.
- One or more application layer protocols.
- Traffic direction.

Restrictions and guidelines

File filtering rules applied to the NFS protocol take effect only on NFSv3 traffic.

File filtering rules applied to the SMB protocol take effect only on SMBv1 and SMBv2 traffic.

The **`logging`** keyword enables the file filtering module to log packet matching events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output file filtering logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view file filtering logs stored on the device, use the **`display logbuffer`** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Create a file filtering policy and enter its view.

   `file-filter policy` *`policy-name`*

3. (Optional.) Configure a description for the file filtering policy.

   `description` *`string`*

   By default, a file filtering policy does not have a description.

4. Create a file filtering rule and enter its view.

   `rule` *`rule-name`*

5. Specify a file type group for the file filtering rule.

   `filetype-group` *`group-name`*

   By default, a file filtering rule does not contain any file type group.

6. Specify the application layer protocols to which the file filtering rule applies.

   `application` { **`all`** | **`type`** { **`ftp`** | **`http`** | **`imap`** | **`nfs`** | **`pop3`** | **`rtmp`** | **`smb`** | **`smtp`** } * }

   By default, no applicable application layer protocols are specified for a file filtering rule.

7. Specify the traffic directions to which the file filtering rule applies.

   `direction` { **`both`** | **`download`** | **`upload`** }

   By default, a file filtering rule applies to upload traffic.

8. Specify the actions to take on matching packets.

```
action { drop | permit } [ logging ]
```

The default action of a file filtering rule is **drop**.

Setting the action for packets with files carrying false extensions

About this task

A packet might contain files that carry false extensions. For example, a file that carries the .exe file extension might actually be a .txt file.

Use this command to specify the action for packets with files carrying false extensions. To perform file filtering inspection based on the real file extension, set the action to **permit**. To discard such packets directly, set the action to **drop**.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the action for packets with files carrying false extensions.

   ```
   file-filter false-extension action { drop | permit }
   ```

   The default action is **permit**, which enables the device to determine the packet processing action based on the real file extension.

Applying a file filtering policy to a DPI application profile

About this task

A file filtering policy must be applied to a DPI application profile to take effect.

A DPI application profile can use only one file filtering policy. If you apply different file filtering policies to the same DPI application profile, only the most recent configuration takes effect.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DPI application profile view.

   ```
   app-profile profile-name
   ```

   For more information about this command, see DPI engine commands in *DPI Command Reference*.

3. Apply a file filtering policy to the DPI application profile.

   ```
   file-filter apply policy policy-name
   ```

   By default, no file filtering policy is applied to the DPI application profile.

Activating file filtering policy and rule settings

About this task

By default, the system will detect whether another configuration change (such as creation, modification, or deletion) occurs within a 20-second interval after a change to the file filtering policy and rule settings:

- If no configuration change occurs within the interval, the system performs an activation operation at the end of the next 20-second interval to make the configuration take effect.
- If a configuration change occurs within the interval, the system continues to periodically detect whether configuration changes occur within next 20-second intervals.

To activate the policy and rule configurations immediately, you can execute the **`inspect activate`** command.

For more information about configuration activation for DPI service modules, see "Configuring DPI engine."

Procedure

1. Enter system view.

   `system-view`

2. Activate file filtering policy and rule settings.

   `inspect activate`

   By default, file filtering policy and rule settings will be activated automatically.

△ **CAUTION:**

This command can cause temporary outage for DPI services. Services based on the DPI services might also be interrupted. For example, security policies cannot control application access.

Applying a DPI application profile to a security policy rule

1. Enter system view.

   `system-view`

2. Enter security policy view.

   security-policy { **ip** | **ipv6** }

3. Enter security policy rule view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Set the rule action to pass.

   `action pass`

   The default rule action is **`drop`**.

5. Use a DPI application profile in the rule.

   profile *app-profile-name*

   By default, no DPI application profile is used in a security policy rule.

File filtering configuration examples

Example: Using a file filtering policy in a security policy

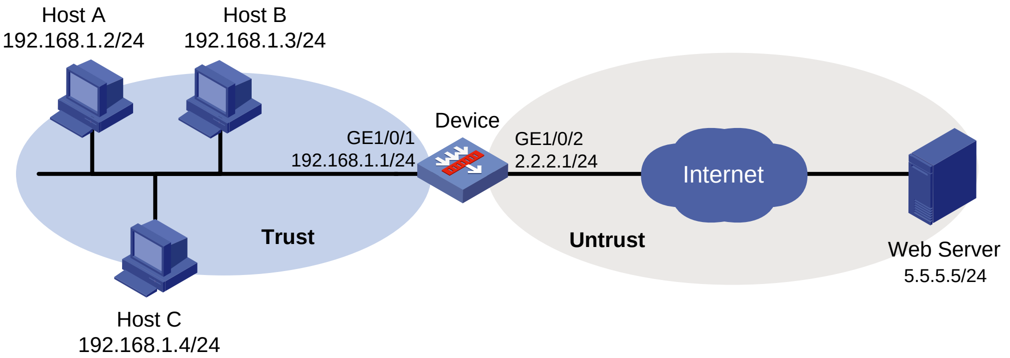
Network configuration

As shown in Figure 1, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

Configure file filtering on the device so the device performs the following operations:

- Blocks files with the **pptx** or **dotx** extension.
- Logs the blocked files.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the Web server, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure file filtering:

   a. Create a file type group named **fg1** and create two file type match patterns to match files with the **pptx** and **dotx** extensions, respectively.

```
[Device] file-filter filetype-group fg1
```

```
[Device-file-filter-fgroup-fg1] pattern 1 text pptx
[Device-file-filter-fgroup-fg1] pattern 2 text dotx
[Device-file-filter-fgroup-fg1] quit
```

b. Create a file filtering rule named **r1** and configure it to drop and log both upload and download HTTP packets that match file type group **fg1**.

```
[Device] file-filter policy p1
[Device-file-filter-policy-p1] rule r1
[Device-file-filter-policy-p1-rule-r1] filetype-group fg1
[Device-file-filter-policy-p1-rule-r1] application type http
[Device-file-filter-policy-p1-rule-r1] direction both
[Device-file-filter-policy-p1-rule-r1] action drop logging
[Device-file-filter-policy-p1-rule-r1] quit
```

5. Configure a DPI application profile and activate the file filtering policy and rule settings:

# Create a DPI application profile named **sec** and apply file filtering policy **p1** to the DPI application profile.

```
[Device] app-profile sec
[Device-app-profile-sec] file-filter apply policy p1
[Device-app-profile-sec] quit
```

# Activate the file filtering policy and rule settings.

```
[Device] inspect activate
```

6. Configure a security policy:

# Create a security policy rule named **trust-untrust**. Configure the rule to apply DPI application profile **sec** to packets from security zone **Trust** to security zone **Untrust with** source subnet address **192.168.1.0/24**.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Verify that the device blocks and logs files that meet the specified criteria. (Details not shown.)

Contents

Configuring anti-virus

About anti-virus

Anti-virus identifies viruses in the application layer of packets based on an up-to-date virus signature library and takes actions to prevent a network from being infected. This feature is typically deployed on a gateway to insulate the internal network from viruses and protect the internal data.

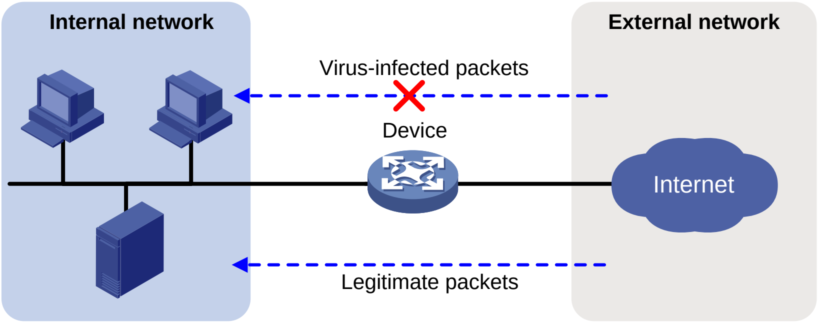
Anti-virus supports inspecting packets transported through FTP, HTTP, IMAP, NFS, POP3, SMB, and SMTP.

Application scenario

As shown in Figure 1, the device is the gateway of an internal network. Internal users access the external network and download data from the external network. The internal server accepts data uploaded by external users.

In this scenario, you can configure anti-virus on the gateway to protect the internal network. Anti-virus inspects incoming packets, permits legitimate packets to pass, and takes actions, such as alert, block, or redirect, on packets containing viruses.

Figure 1 Anti-virus application scenario



Terminology

Virus signature

A virus signature is a character string that uniquely identifies a specific virus. The virus signature library contains the predefined virus signatures.

MD5 rules

An MD5 rule is generated by the system based on the virus signatures in the virus signature library to identify virus-infected files.

Signature exception

Typically, anti-virus takes anti-virus actions on packets matching virus signatures. If a virus proves to be a false alarm, you can set the virus signature as a signature exception. Packets matching the signature exception are permitted to pass.

Application exception

Typically, anti-virus action is protocol specific and applies to all applications carried by the protocol. To take a different action on an application, you can set the application as an exception and specify a different anti-virus action for the application. Application exceptions use application-specific actions and the other applications use protocol-specific actions. For example, the anti-virus action for HTTP is alert. To block the games carried by HTTP, you can set the games as application exceptions and specify the block action for them.

MD5 exception

If false positives occur for a virus, you can set the MD5 value of the virus as an MD5 exception. The device will permit subsequent packets matching the MD5 exception to pass.

You can get the MD5 value of the virus through the threat log.

Anti-virus action

Anti-virus actions apply to the packets that match virus signatures. The actions include the following types:

- **alert**—Permits matching packets and generates logs.
- **block**—Blocks matching packets and generates logs.
- **redirect**—Redirects matching HTTP connections to a URL and generates logs. The redirection is applicable to only uploading connections.

The generated anti-virus logs can be sent to the device information center or to designated recipients by email.

Virus detection methods

The device supports the following virus detection methods:

- **Virus signature-based detection**—The device matches packets against virus signatures in the virus signature library, and determines that a packet contains viruses if a match is found.
- **MD5 rule-based detection**—The device generates an MD5 hash value for a file to be inspected and compares the value with the system-defined MD5 rules. If a match is found, the file is identified to be virus-infected.

Anti-virus mechanism

Anti-virus takes effect after you apply an anti-virus policy to a DPI application profile and use the DPI application profile in a security policy rule.

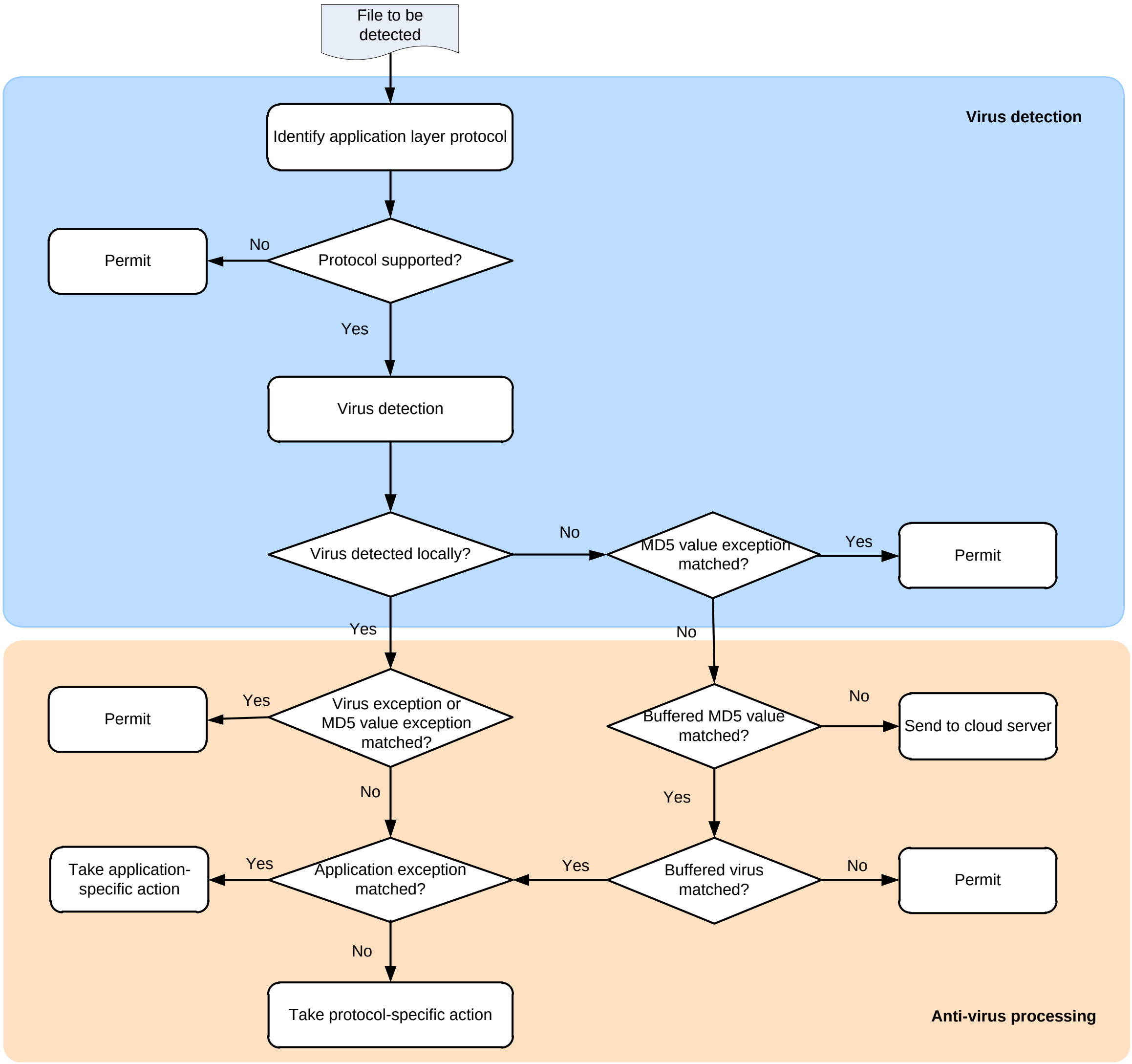
As shown in Figure 2, upon receiving a packet, the anti-virus device performs the following operations:

1. The device identifies whether the anti-virus supports the application layer protocol of the packet.
   - If not, the device permits the packet to pass without virus detection.
   - If yes, the device compares the packet with the virus signatures and MD5 rules.
2. If a matching signature or MD5 rule is found, the device determines if the matching signature is an exception.
   - If yes, the device permits the packet to pass.
   - If not, the device examines whether the application is an exception.
     - If the application is an exception, the device takes the application-specific action (alert, block, or permit).
     - If the application is not an exception, the device takes the protocol-specific action (alert, block, or redirect).
3. If no matching signature or MD5 rule is found, the device examines whether the MD5 value of the file in the packet matches an MD5 value exception.
   - If yes, the device permits the packet to pass.

- If not, the device examines whether the MD5 value of the file in the packet matches a buffered MD5 value in the anti-virus buffer.
  - If a buffered MD5 value labeled as virus is found, the device determines if the application is an exception. If yes, the device takes the application-specific action (alert, block, or permit). If not, the device takes the protocol-specific action (alert, block, or redirect).
  - If a buffered MD5 value labeled as non-virus is found, the device permits the packet to pass.
  - If no buffered MD5 value is found, the device sends the MD5 value of the file to the cloud server for future virus detection.

  After the virus detection, the device saves the detection results returned from the cloud server to the anti-virus buffer so the virus detection for subsequent packets can be performed locally.

Figure 2 Anti-virus mechanism


File to be detected
Virus detection
Identify application layer protocol
Permit
No
Protocol supported?
Yes
Virus detection
Virus detected locally?
No
MD5 value exception matched?
Yes
Permit
Yes
No
Permit
Yes
Virus exception or MD5 value exception matched?
Buffered MD5 value matched?
No
Send to cloud server
No
Yes
Take application-specific action
Yes
Application exception matched?
Yes
Buffered virus matched?
No
Permit
No
Take protocol-specific action
Anti-virus processing

Virus signature library management

The device inspects packets for viruses based on the virus signature library. You can update the virus signature library to the latest version or roll it back to the previous version or the factory default version.

Updating the virus signature library

The following methods are available for updating the virus signature library:

- Automatic update.

  The device automatically and periodically downloads the most up-to-date virus signature file to update the signature library.
- Triggered update.

  The device downloads the most up-to-date virus signature file to update the signature library immediately after you trigger the operation.
- Manual update.

  Use this method when the device cannot obtain the virus signature file automatically.

  You must manually download the most up-to-date virus signature file and then use the downloaded file to update the signature library.

Rolling back the virus signature library

If the false alarm rate is high or abnormal situations frequently occur, you can roll back the virus signature library to the previous version or to the factory default version.

Licensing requirements

Before using the anti-virus feature, purchase and correctly install a license on the device. If the license expires, the anti-virus feature is still available but you can no longer update the virus signature library, perform MD5 value cloud query, or collaborate with sandbox to block packets. For more information about licenses, see *Fundamentals Configuration Guide*.

Restrictions and guidelines: Anti-virus

Anti-virus supports inspecting packets transported through FTP, HTTP, HTTPS, IMAP, IMAPS, NFS, POP3, POP3S, SMB, SMTP, and SMTPS. To inspect packets transported through HTTPS, IMAPS, POP3S, and SMTPS, you must use anti-virus together with SSL proxy. For information about SSL proxy, see "Configuring proxy policy."

Anti-virus tasks at a glance

To configure anti-virus, perform the following tasks:

1. Configuring an anti-virus policy
2. Configuring MD5 value-based anti-virus cloud query
3. Specifying a parameter profile for an anti-virus action
4. Applying an anti-virus policy to a DPI application profile
5. (Optional.) Activating anti-virus policy settings
6. Applying a DPI application profile to a security policy rule
7. Managing the virus signature library

Configuring an anti-virus policy

About this task

An anti-virus policy defines the virus detection criteria, anti-virus actions, virus signature exceptions, and application exceptions.

The virus signatures in the virus signature library are available to all anti-virus policies on the device.

The device supports sending the alarm message defined in the warning parameter profile applied to the policy. If an endpoint user visits a virus-infected website, the device will display the alarm message on the user's browser. For more information about configuring a warning parameter profile, see "Configuring DPI engine."

Restrictions and guidelines

Anti-virus supports only NFSv3 of the NFS protocol, and SMBv1 and SMBv2 of the SMB protocol.

The **logging** keyword enables the anti-virus module to log the packet matching events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output anti-virus logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view anti-virus logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create an anti-virus policy and enter its view.

   anti-virus policy *policy-name*

   A default anti-virus policy named **default** exists. The default anti-virus policy cannot be modified or deleted.

3. (Optional.) Configure a description for the anti-virus policy.

   description *text*

4. Configure anti-virus for an application layer protocol.

   inspect { **ftp** | **http** | **imap** | **nfs** | **pop3** | **smb** | **smtp** } **direction** { **both** | **download** | **upload** } [**cache-file-size** *file-size*] **action** { **alert** | **block** | **redirect** }

   By default, the device performs virus detection on upload and download packets for FTP, HTTP, IMAP, NFS, and SMB, on download packets for POP3, and on upload packets for SMTP. The anti-virus action for FTP, HTTP, NFS, and SMB is **block** and for IMAP, SMTP, and POP3 is **alert**. The maximum size for the file that can be cached for virus detection is 1 MB. The **direction** keyword is not available for the POP3 and SMTP protocols because the POP3 protocol supports only the download direction and the SMTP protocol supports only the upload direction.

5. (Optional.) Apply a warning parameter profile to an anti-virus policy and enable sending the alarm message defined in the profile.

   warning parameter-profile *profile-name*

   By default, no warning parameter profile is applied and the device does not support the sending of alarm messages.

   The alarm message sending takes effect only when the HTTP protocol and the **block** action are configured for virus detection.

6. (Optional.) Set a signature as a signature exception.

   exception signature *signature-id*

7. (Optional.) Set an application as an application exception and specify an anti-virus action for the application exception.

   exception application *application-name* **action** { **alert** | **block** | **permit** }

8. (Optional.) Set an MD5 value as an MD5 exception.

   exception md5 *md5-value*

9. Enable the virus signatures at and above a severity level.

   signature severity { **critical** | **high** | **medium** } **enable**

   By default, virus signatures of all severity levels are enabled.

Configuring MD5 value-based anti-virus cloud query

About this task

You can enable MD5 value-based anti-virus cloud query in an anti-virus policy. If no virus is found in the file, the device will send the MD5 value of the file to the cloud server for cloud query. The cloud server determines whether the MD5 value is a virus and returns the result to the device so appropriate action can be taken. The anti-virus module will save the result returned from the cloud server to the anti-virus buffer so the virus detection for subsequent packets can be performed locally.

For more information about the cloud query server, see "Configuring DPI engine."

Restrictions and guidelines

MD5 value-based anti-virus cloud query is available only for the following protocols:

- HTTP.
- IMAP.
- NFS. Only the NFS read operation is supported.
- POP3.
- SMTP.

Procedure

1. Enter system view.

   system-view

2. Specify the cloud query server.

   inspect cloud-server *host-name*

3. (Optional.) Set the anti-virus cache size.

   anti-virus cache size *cache-size*

   By default, the anti-virus cache can cache a maximum of 100000 entries.

4. (Optional.) Set the minimum cache period for an anti-virus MD5 entry.

   anti-virus cache min-time *value*

   By default, the minimum cache period of an anti-virus MD5 entry is 10 minutes.

5. Enter anti-virus policy view.

   anti-virus policy *policy-name*

6. Enable MD5 value-based anti-virus cloud query.

   cloud-query enable

   By default, MD5 value-based anti-virus cloud query is disabled.

Specifying a parameter profile for an anti-virus action

About this task

Before you can specify a parameter profile for an anti-virus action, configure the parameter profile in the DPI engine. For more information, see "Configuring DPI engine."

A parameter profile defines the parameters for executing an action. For example, you can configure parameters such as the email server address and email recipients in the email parameter profile, and then apply the profile to the email action.

If no parameter profile is specified for an anti-virus action, or if the specified parameter profile does not exist, the default parameter settings of the action are used.

Procedure

1. Enter system view.

   system-view

2. Specify a parameter profile for an anti-virus action.

   anti-virus { **email** | **logging** | **redirect** } **parameter-profile** *profile-name*

   By default, no parameter profile is specified for an anti-virus action.

Applying an anti-virus policy to a DPI application profile

About this task

The DPI application profile is a template for configuring DPI security services. For an anti-virus policy to take effect, you must apply it to a DPI application profile.

A DPI application profile can use only one anti-virus policy. If you apply different anti-virus policies to the same DPI application profile, only the most recent configuration takes effect.

Procedure

1. Enter system view.

   system-view

2. Enter DPI application profile view.

   app-profile *profile-name*

   For more information about this command, see DPI engine commands in *DPI Command Reference*.

3. Apply an anti-virus policy to the DPI application profile.

   anti-virus apply policy *policy-name* **mode** { **alert** | **protect** }

   By default, no anti-virus policy is applied to a DPI application profile.

Activating anti-virus policy settings

About this task

By default, the system will detect whether another configuration change (such as creation, modification, or deletion) occurs within a 20-second interval after a change to the anti-virus policy and rule settings:

- If no configuration change occurs within the interval, the system performs an activation operation at the end of the next interval (40 seconds later) to make the configuration take effect.
- If a configuration change occurs within the interval, the system continues to periodically check whether a configuration change occurs within the interval.

To activate the policy and rule configurations immediately, you can execute the **`inspect activate`** command.

Procedure

1. Enter system view.
   `system-view`
2. Activate anti-virus policy settings.
   `inspect activate`
   By default, anti-virus policy settings will be activated automatically.

△ **CAUTION:**

This command can cause temporary outage for the DPI service and other services based on DPI. For example, security policies cannot perform application access control.

Applying a DPI application profile to a security policy rule

1. Enter system view.
   `system-view`
2. Enter security policy view.
   `security-policy` { **`ip`** | **`ipv6`** }
3. Enter security policy rule view.
   `rule` { *`rule-id`* | [*`rule-id`*] **name** *`rule-name`* }
4. Set the rule action to pass.
   `action pass`
   The default rule action is **`drop`**.
5. Use a DPI application profile in the rule.
   `profile` *`app-profile-name`*
   By default, no DPI application profile is used in a security policy rule.

Managing the virus signature library

As viruses constantly increase and change, you must update the virus signature library in time. You can also roll back the virus signature library.

Restrictions and guidelines

- Do not delete the **/dpi/** folder in the root directory of the storage medium.
- Do not perform virus signature update and rollback when the device's free memory is below the normal state threshold. For more information about device memory thresholds, see device management in *Fundamentals Configuration Guide*.
- For successful automatic and immediate signature update, make sure the device can resolve the domain name of the NSFOCUS website into an IP address through DNS. For more information about DNS, see DNS configuration in *Layer 3—IP Services Configuration Guide.*

Scheduling automatic virus signature library update

About this task

You can schedule automatic virus signature library update if the device can access the signature database services on the NSFOCUS website. The device periodically obtains the latest signature file from the NSFOCUS website to update its local signature library according to the update schedule.

Procedure

1. Enter system view.

   system-view

2. Enable automatic virus signature library update and enter automatic virus signature library update configuration view.

   anti-virus signature auto-update

   By default, automatic virus signature library update is disabled.

3. Schedule the update time.

   update schedule { **daily** | **weekly** { **fri** | **mon** | **sat** | **sun** | **thu** | **tue** | **wed** } } **start-time** *time* **tingle** *minutes*

   By default, the device updates the virus signature library at a random time between 02:01:00 and 04:01:00 every day.

Triggering an immediate automatic virus signature library update

About this task

Anytime you find a new release of virus signature file on the NSFOCUS website, you can trigger the device to immediately and automatically update the virus signature library.

Procedure

1. Enter system view.

   system-view

2. Trigger an immediate virus signature library update.

   anti-virus signature auto-update-now

Manually updating the virus signature library

About this task

If the device cannot access the signature database services on the NSFOCUS website, use one of the following methods to manually update the virus signature library:

- **Local update**—Updates the virus signature library by using the locally stored virus signature file.

  Store the update file on the master device for successful signature library update.
- **FTP/TFTP update**—Updates the virus signature library by using the virus signature file stored on an FTP or TFTP server.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Manually update the virus signature library.
   ```
   anti-virus signature update file-path
   ```

△ **CAUTION:**

The NSFOCUS website provides different signature libraries for devices with different memory sizes and software versions. You must obtain the signature library that is suitable for your device. If your device has a small memory (8 GB or less) but you choose a signature library that is for a large memory (more than 8 GB), the signature update might result in device anomaly.

Rolling back the virus signature library

About this task

If a virus signature library update causes abnormal situations or a high false alarm rate, you can roll back the virus signature library.

Before rolling back the virus signature library, the device backs up the current signature library as the previous version. For example, the previous version is V1 and the current version is V2. If you perform a rollback to the previous version, version V1 becomes the current version and version V2 becomes the previous version. If you perform a rollback to the previous version again, version V2 becomes the current version and version V1 becomes the previous version.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Roll back the virus signature library.
   ```
   anti-virus signature rollback { factory | last }
   ```

Display and maintenance commands for anti-virus

Execute the **display** commands in any view.

Task	Command
Display anti-virus cache information.	**display anti-virus cache** [**slot** *slot-number*]
Display virus signature information.	**display anti-virus signature** [[*signature-id*] | [**severity** { **critical** | **high** | **low** | **medium** }]]
Display virus signature family information.	**display anti-virus signature family-info**
Display virus signature library information.	**display anti-virus signature library**

Task	Command
Display anti-virus statistics.	**display anti-virus statistics** [**policy** *policy-name*] [**slot** *slot-number*]

Anti-virus configuration examples

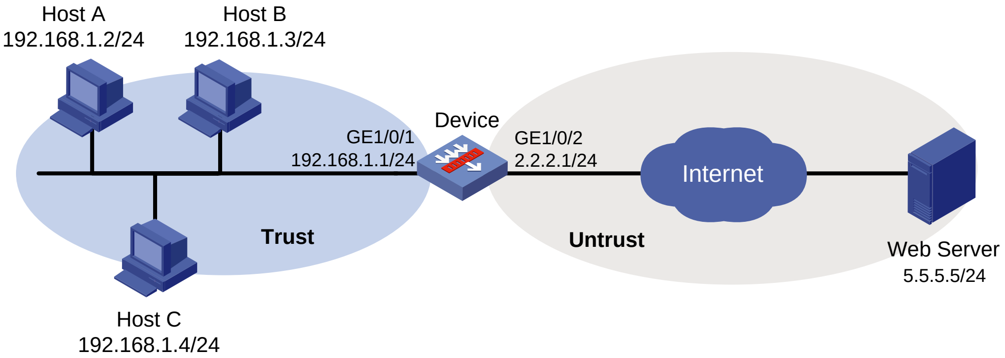
Example: Using the default anti-virus policy in a security policy

Network configuration

As shown in Figure 3, the device connects the LAN and the Internet. The LAN resides in security zone **Trust** and the Internet resides in security zone **Untrust**.

Configure the device to use the default anti-virus policy for virus detection and prevention.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a DPI application profile and activate the anti-virus policy settings:

# Apply the default anti-virus policy to DPI application profile **sec** and set the policy mode to **protect**.

```
[Device] app-profile sec
[Device-app-profile-sec] anti-virus apply policy default mode protect
[Device-app-profile-sec] quit
```

# Activate the anti-virus policy settings.

```
[Device] inspect activate
```

5. Configure a security policy:

# Create a security policy rule named **trust-untrust**. Configure the rule to apply DPI application profile **sec** to packets from security zone **Trust** to security zone **Untrust** with source subnet address **192.168.1.0/24**.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

# Verify that the device can use the default anti-virus policy to detect and prevent known viruses. (Details not shown.)

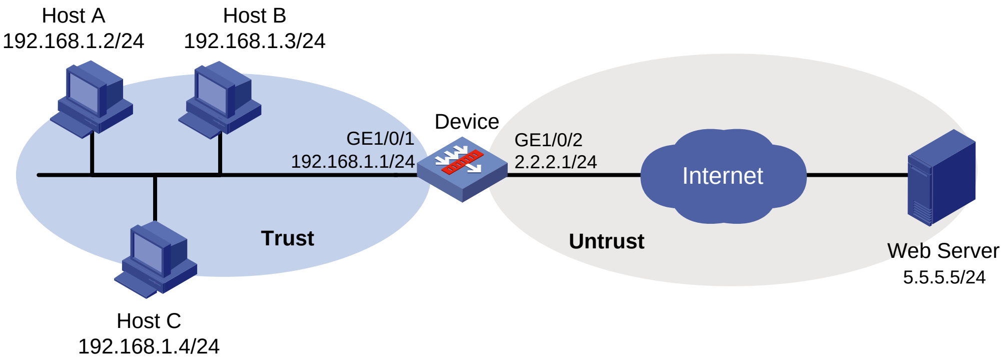
Example: Using a user-defined anti-virus policy in a security policy

Network configuration

As shown in Figure 4, the device connects the LAN and the Internet. The LAN resides in security zone **Trust** and the Internet resides in security zone **Untrust**.

Configure the device to use a user-defined anti-virus policy for virus detection and prevention. In the user-defined anti-virus policy, set virus signature 2 as a signature exception and set the **139Email** application as an application exception.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Create anti-virus policy **antivirus1**, set virus signature 2 as a signature exception, set the **139Email** application as an application exception, and specify **alert** as the anti-virus action for the application exception.

   ```
   [Device] anti-virus policy antivirus1
   [Device-anti-virus-policy-antivirus1] exception signature 2
   [Device-anti-virus-policy-antivirus1] exception application 139Email action alert
   [Device-anti-virus-policy-antivirus1] quit
   ```

5. Configure a DPI application profile and activate the anti-virus policy settings:

   # Apply anti-virus policy **antivirus1** to DPI application profile **sec** and set the policy mode to **protect**.

   ```
   [Device] app-profile sec
   [Device-app-profile-sec] anti-virus apply policy antivirus1 mode protect
   [Device-app-profile-sec] quit
   ```

   # Activate the anti-virus policy settings.

   ```
   [Device] inspect activate
   ```

6. Configure a security policy:

   # Create a security policy rule named **trust-untrust**. Configure the rule to apply DPI application profile **sec** to packets from security zone **Trust** to security zone **Untrust** with source subnet address **192.168.1.0/24**.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   [Device-security-policy-ip-10-trust-untrust] profile sec
   [Device-security-policy-ip-10-trust-untrust] quit
   ```

   # Activate rule matching acceleration.

   ```
   [Device-security-policy-ip] accelerate enhanced enable
   [Device-security-policy-ip] quit
   ```

Verifying the configuration

# Verify that the anti-virus policy is correctly configured. (Details not shown.)

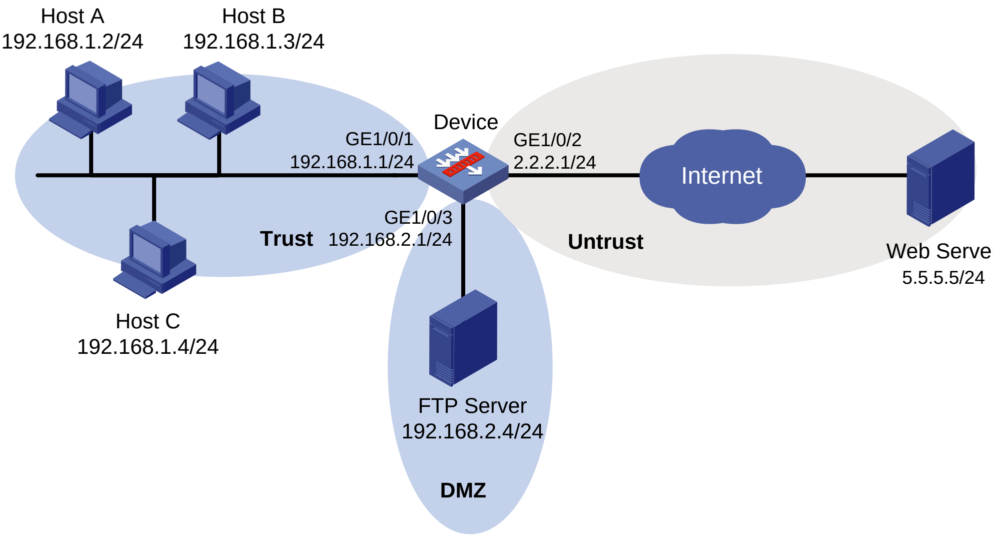
Example: Manually updating the virus signature library

Network configuration

As shown in Figure 5, LAN users in security zone **Trust** can access the Internet resources in security zone **Untrust** and the FTP server in security zone **DMZ**. The username and password for logging in to the FTP server are **anti-virus** and **123**, respectively. The latest virus signature file **anti-virus-1.0.8-encrypt.dat** is stored in the root directory on the FTP server.

Manually update the virus signature library on the device by using the latest virus signature file on the FTP server.

Figure 5 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

```
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
[Device-security-zone-DMZ] quit
```

4. Configure a security policy:

   # Create a security policy rule named **trust-untrust** to permit traffic sent from security zone **Trust** to security zone **Untrust**.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   [Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-10-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-10-trust-untrust] action pass
   [Device-security-policy-ip-10-trust-untrust] quit
   ```

   # Create a security policy rule named **trust-dmz** to permit traffic sent from security zone **Trust** to security zone **DMZ**.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-dmz
   [Device-security-policy-ip-11-trust-dmz] source-zone trust
   [Device-security-policy-ip-11-trust-dmz] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-11-trust-dmz] destination-zone dmz
   [Device-security-policy-ip-11-trust-dmz] action pass
   [Device-security-policy-ip-11-trust-dmz] quit
   ```

   # Create a security policy rule named **downloadlocalout** to permit traffic sent from security zone **Local** to security zone **DMZ**. Thus, the internal hosts can access the FTP server to obtain the virus signature files.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name downloadlocalout
   [Device-security-policy-ip-12-downloadlocalout] source-zone local
   [Device-security-policy-ip-12-downloadlocalout] destination-zone dmz
   [Device-security-policy-ip-12-downloadlocalout] destination-ip-subnet 192.168.2.0
   24
   [Device-security-policy-ip-12-downloadlocalout] application ftp
   [Device-security-policy-ip-12-downloadlocalout] application ftp-data
   [Device-security-policy-ip-12-downloadlocalout] action pass
   [Device-security-policy-ip-12-downloadlocalout] quit
   ```

   # Activate rule matching acceleration.

   ```
   [Device-security-policy-ip] accelerate enhanced enable
   [Device-security-policy-ip] quit
   ```

5. Update the virus signature library by using the virus signature file **anti-virus-1.0.8-encrypt.dat** on the FTP server.

   ```
   [Device] anti-virus signature update
   ftp://anti-virus:123@192.168.2.4/anti-virus-1.0.8-encrypt.dat
   ```

Verifying the configuration

# Verify that the virus signature library is successfully updated.

```
<Device> display anti-virus signature library
```

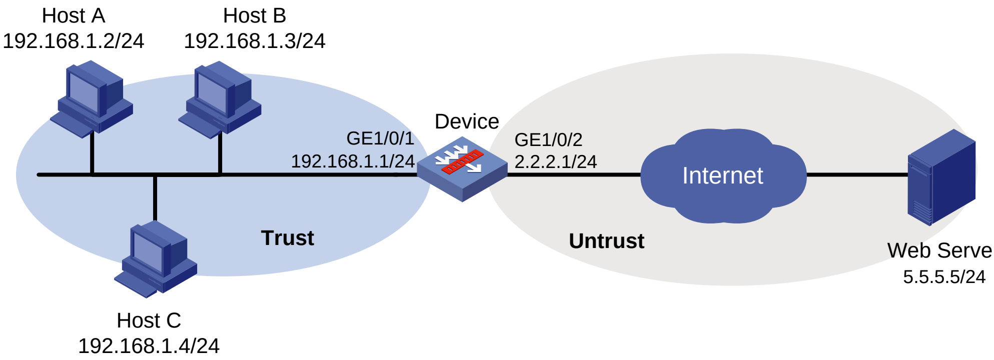
Example: Configuring automatic virus signature library update

Network configuration

As shown in Figure 6, LAN users in security zone **Trust** can access Internet resources in security zone **Untrust**.

Configure the device to automatically update the virus signature library at a random time between 08:30 a.m. and 09:30 a.m. every Saturday.

Figure 6 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure a DNS server to resolve the domain name of the official website into an IP address.

   ```
   [Device] dns server 10.72.66.36
   ```

5. Configure a security policy:

   # Create a security policy rule named **trust-untrust** to permit traffic sent from security zone Trust to security zone **Untrust**.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-10-trust-untrust] source-zone trust
   ```

```
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] quit
```

# Create a security policy rule named **downloadlocalout** to permit traffic sent from security zone Local to security zone **Untrust**. Thus, the internal hosts can access official website to obtain the virus signature files.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name downloadlocalout
[Device-security-policy-ip-11-downloadlocalout] source-zone local
[Device-security-policy-ip-11-downloadlocalout] destination-zone untrust
[Device-security-policy-ip-11-downloadlocalout] action pass
[Device-security-policy-ip-11-downloadlocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

6. Configure automatic virus signature library update:

# Enable automatic virus signature library update and configure the device to perform automatic update at a random time between 08:00 a.m. and 10:00 a.m. every Saturday.

```
[Device] anti-virus signature auto-update
[Device-anti-virus-autoupdate] update schedule weekly sat start-time 9:00:00 tingle
60
[Device-anti-virus-autoupdate] quit
```

Verifying the configuration

# Verify that the virus signature library is updated as scheduled.

```
<Device> display anti-virus signature library
```

Contents

Configuring the data analysis center

About the data analysis center

The data analysis center collects and analyzes log data for services and provides the analysis results in various forms of reports through the Web interface. It supports log data storage, traffic monitoring, and report analysis. This feature allows you to learn about the service traffic statistics and the network security status, helping you make decisions when customizing service policies.

Log data storage and analysis

The data analysis center collects log data from various service modules for central analysis and reporting. You can store the log data in a hard disk, a USB drive, or the memory according to the storage priority in descending order. If the storage media of a higher priority is not available or its space is full, the data are stored in the storage media of a lower priority.

Traffic monitoring

The data analysis center generates real-time traffic trend and statistics reports from various perspectives, such as user, application, and IP address. These reports help you monitor the network traffic, locate network vulnerabilities, and secure the network against potential attacks.

Reporting

The data analysis center can generate the multiple types of reports for you to understand the information such as service statistics, device running status, and network security status.

Restrictions and guidelines: Data analysis center configuration

You can configure the data analysis center at the CLI. The reports generated by the data analysis center are available only in the Web interface.

Data analysis center tasks at a glance

To configure the data analysis center, perform the following tasks:

- Enabling log collection
- Enabling real-time log display
- Enabling real-time traffic statistics collection
- Configuring the email server
- Configuring report subscription
- Configuring report export
    a. Configuring a report export template
    b. Configuring the report contents
       Choose one of the following tasks:

    - Configuring an LB link statistics report
    - Configuring an LB virtual server statistics report
  - **c.** Configuring report export parameters
- Configuring data storage limits for a service

Enabling log collection

About this task

The log collection feature enables the data analysis center to collect the log messages of specific services and extracts the data for summarization and analysis. You can see the relevant data analysis information in the dashboard and monitor pages of the Web interface.

Restrictions and guidelines

To collect the log messages for the traffic service, first enable the session statistics collection and then enable the log collection. For more information about the session statistics collection, see session management in *Security Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enable the log collection for a service.

```
dac log-collect service service-type service-name enable
```

By default, the log collection status for each service varies by service setting when the service module is registered to the DAC.

Enabling real-time log display

About this task

With this feature enabled for a service, the data analysis center will send the service log messages to the Web interface in real time. You can see the real-time logs on the Web interface without refreshing the log lists manually.

Restrictions and guidelines

The real-time log display setting for a service takes effect only after the log collection for the service is enabled by the `dac log-collect enable` command.

Procedure

1. Enter system view.

```
system-view
```

2. Enable the real-time log display.

```
dac log-display service service-type service-name enable
```

The log collection status for each service varies by service setting when the service module is registered to the DAC.

Enabling real-time traffic statistics collection

About this task

The data analysis center can collect the user and application traffic statistics in real time and send the statistics result to the Web interface.

Restrictions and guidelines

Enable real-time traffic statistics collection with caution when large-volume service traffic exists. This feature is CPU intensive.

To collect the traffic statistics in real time, you must first enable the session statistics collection. For more information about the session statistics collection, see session management in *Security Configuration Guide*.

Procedure

1. Enter system view.

system-view

2. Enable real-time traffic statistics collection.

dac traffic-statistic { **application** | **user** } **enable** [**verbose**]

By default, the real-time collection of traffic statistics is disabled.

Configuring the email server

About this task

The report subscription and report export features require an email server to function correctly. The email server will send the reports to the subscribers or export the reports to the specified mail box.

Procedure

1. Enter system view.

system-view

2. Specify the email server address.

dac email-server server-address *address-string*

By default, no email server is specified for the data analysis center.

3. Specify the email sender address.

dac email-server sender *address-string*

By default, the email sender address is not specified.

4. (Optional.) Configure email client authentication.

a. Enable email client authentication.

dac email-server client-authentication enable

By default, email client authentication is disabled.

b. Specify the username for email client authentication.

dac email-server username *username*

By default, no username is specified for email client authentication.

c. Specify the password for email client authentication.

dac email-server password { **cipher** | **simple** } *string*

By default, no password is specified for email client authentication.

d. Enable secure transmission of client authentication credentials.

```
dac email-server secure-authentication enable
```

By default, secure transmission of client authentication credentials is disabled.

Configuring report subscription

About this task

The report subscription allows the device to generate and send periodic reports to the subscriber mail box.

By default, the daily report is sent during the least busy hours (1 am. to 5 am.) and the monthly report of the previous month is sent on the first day of each month. The report sending time cannot be changed.

The following types of reports are supported:

- **Summary report**—Displays summarized service traffic statistics collected over a time range.
- **Comparison report**—Provides comparison of service traffic statistics collected over two time ranges that contain the same number of days.
- **Intelligent report**—Provides intelligent analysis of users' work efficiency, data leakage, and turnover risks based on their network access behaviors.
- **Integrated report**—Illustrates the overall device operational and network security status based on analysis of critical service statistics.

Prerequisites

For the subscribers to receive the reports, you must configure the email server.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the subscription parameters for a report type.

   ```
   dac report type { comparison | integrated | intelligent | summary }
   subscriber mail-address [ language { chinese | english } ]
   ```

   By default, no report subscription parameters are configured.

Configuring report export

About report export

Report export periodically or immediately exports statistics reports for specified services. The following types of export methods are available:

- **Automatic export**—Exports periodic reports to the report destinations as scheduled. The report contents are defined in the report template.
- **Manual export**—Exports reports immediately after you define the statistics contents and time range of the data.

Restrictions and guidelines

Manual report export is supported only on the Web interface of the product.

Report export is available only for LB services. For more information about LB, see load balancing in *Load Balancing Configuration Guide*.

Prerequisites

For the subscribers to receive the reports, you must configure the email server.

Configuring a report export template

About this task

In a report export template, you can define the following items:

- Report language.
- Statistics contents in the report.

Procedure

1. Enter system view.
   system-view
2. Create a report export template and enter its view.
   dac report export template *template-name*
3. (Optional.) Specify the language used in exported reports.
   language { **chinese** | **english** }
   By default, Chinese is used.

Configuring an LB link statistics report

1. Enter system view.
   system-view
2. Enter the view of an existing report export template.
   dac report export template *template-name*
3. Create the LB link statistics report view and enter the veiw.
   export-service lb-link
4. Specify an LB link for the LB link statistics report.
   statistics link *name*
   By default, no LB link is specified for the LB link statistics report.
5. Specify the contents for the LB link statistics report.
   statistics content { **abnormal-flow** | **app** | **connection-count** | **connection-rate** | **delay** | **packet-loss** | **stability** }*
   By default, no content is specified for the LB link statistics report.

Configuring an LB virtual server statistics report

1. Enter system view.
   system-view
2. Enter the view of an existing report export template.
   dac report export template *template-name*
3. Create the LB virtual server statistics report view and enter the view.
   export-service lb-virtual-server
4. Specify an LB virtual server for the LB virtual server statistics report.

statistics virtual-server *name*

By default, no LB virtual server is specified for the LB virtual server statistics report.

5. Specify the content for the LB virtual server statistics report.

   statistics content class

   By default, no content is specified for the LB virtual server statistics report.

Configuring report export parameters

About this task

You can define the type of the periodic report, report template, and the destination to which the reports are exported.

Procedure

1. Enter system view.

   system-view

2. Configure report export parameters.

   dac report export period { **day** | **hour** | **month** | **quarter** | **week** | **year** } **template** *template-name* [**mail-address** *mail-address*]

   By default, no report export parameters are configured.

Configuring data storage limits for a service

About this task

Perform this task to set the storage time limit, storage space usage limit, and the storage limit-violated action for a service.

The data analysis center periodically checks the data of each service to determine if the storage time or storage space usage limit is exceed.

- If a storage limit is exceeded and the action is **delete**, the system deletes the expired or the oldest service data. A log will be generated to report the event.
- If a storage limit is exceeded and the action is **log-only**, the system generates a log message. New data will not be saved.

Procedure

1. Enter system view.

   system-view

2. Set the storage time limit, storage space usage limit, or the storage limit-triggered action for a service.

   dac storage service *service-type service-name* **limit** { **hold-time** *time-value* | **usage** *usage-value* | **action** { **delete** | **log-only** } }

   By default:

   - The service data can be saved for a maximum of 365 days.
   - The data of each service can occupy up to 20% of the total storage space.
   - If the storage time or storage space usage limit is exceeded, the system deletes the expired or the oldest data.

Display and maintenance commands for data analysis center

Execute the **display** commands in any view.

Task	Command
Display the email server configuration.	**display dac email-server**
Display the log collection configuration for a service.	**display dac log-collect { all | service** *service-type service-name* **}**
Display the configuration of the real-time log display.	**display dac log-display { all | service** *service-type service-name* **}**
Display the report subscription information.	**display dac report** [**comparison** | **integrated** | **intelligent** | **summary**]
Display report export configuration.	**display dac report export**
Display report export templates.	**display dac report export template**
Display the service storage limit settings.	**display dac storage** [*service-type service-name*]
Display the configuration of the real-time traffic statistics collection.	**display dac traffic-statistic** [**application** | **user**]

Contents

Configuring the proxy policy

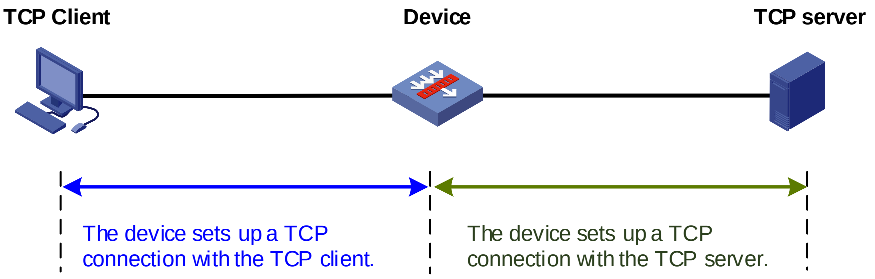
About the proxy policy

The proxy policy enables the device to proxy TCP or SSL connections between clients and servers and implement deep packet inspection and audit on the traffic for high security.

TCP proxy

As shown in Figure 1, the device implements TCP proxy for traffic between the TCP client and server. After receiving traffic from the TCP client to the server, the device becomes a TCP proxy. It proxies the server to establish a connection with the client and proxies the client to establish a connection with the server, respectively. The client and server exchange data through the two TCP connections. This mechanism provides TCP-layer isolation between the TCP client and server and helps effectively block unauthorized access attempts and malicious attacks.

Figure 1 TCP proxy


TCP Client
Device
TCP server
The device sets up a TCP connection with the TCP client.
The device sets up a TCP connection with the TCP server.

SSL proxy

About this task

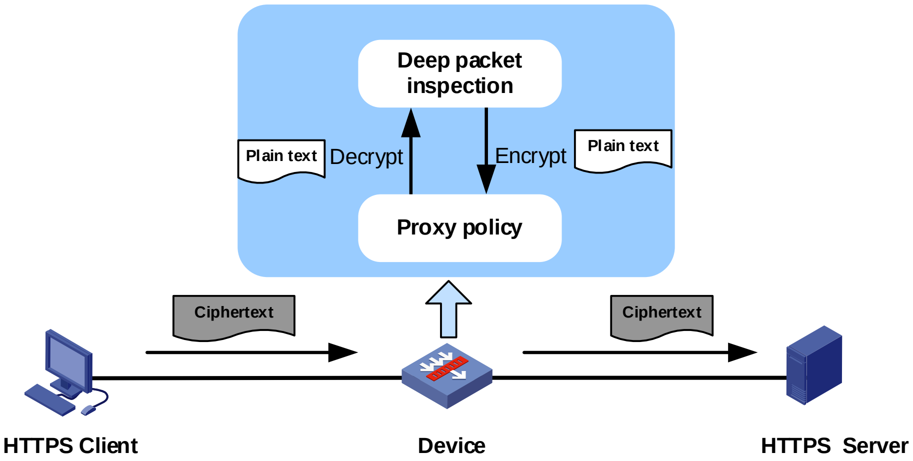
SSL proxy enables the device to decrypt SSL traffic and perform deep packet inspection on the traffic.

As shown in Figure 2, the device implements SSL decryption on HTTPS traffic from the HTTPS client to the HTTPS server. After receiving HTTP traffic from the HTTPS client to the server, the device becomes an SSL proxy. It proxies the server to complete the SSL handshake and establish an SSL connection with the client and proxies the client to establish an SSL connection with the server. When the HTTPS client and server exchange data, the device performs the following operations:

1. Decrypts the HTTPS traffic.
2. Performs deep packet inspection and audit on the traffic according to the DPI service configuration.

   For more information about DPI services, see *DPI Configuration Guide*.
3. Re-encrypts the traffic and send it to the server or client.

Figure 2 SSL proxy



Application scenarios

The SSL decryption can be used in the following scenarios:

- **Protecting internal clients**—The device is deployed at the exit of the network where the internal clients are. When the internal clients access an external server, the device acts as a proxy server to decrypt the packets and perform deep packet inspection on the decrypted packets. It protects the internal clients from being attacked by external malicious websites. In this scenario, the device requires imported SSL decryption certificates to establish SSL connections with the clients. For more information about the SSL decryption certificate, see "SSL decryption certificate."
- **Protecting internal servers**—The device is deployed at the entrance of the network where the internal servers are. When the external clients access an internal server, the device acts as a proxy server to decrypt the packets and perform deep packet inspection on the decrypted packets. It protects the internal servers from being attacked by external malicious traffic. In this scenario, the device requires imported internal server certificates to establish SSL connections with the clients. For more information about the internal server certificate, see "Internal server certificate."

For more information about DPI, see "DPI overview."

Basic concepts

SSL decryption certificate

The device supports a maximum of one trusted SSL decryption certificate and one untrusted SSL decryption certificate, both of which are CA certificates that must be manually imported to the device.

- **Trusted SSL decryption certificate**—SSL decryption certificate that is trusted by clients. The device uses the trusted SSL decryption certificate to sign proxy server certificates for servers whose certificates are verified as legitimate. The trusted SSL decryption certificate must be installed on the client browser so the client can trust the proxy server certificates signed by it.
- **Untrusted SSL decryption certificate**—SSL decryption certificate that clients do not trust. The device uses the untrusted SSL decryption certificate to sign proxy server certificates for servers whose certificates are verified as illegitimate.

Proxy server certificate

In the scenario of protecting internal clients, the device acts as an SSL proxy server to check whether an external server is trusted. After the identity verification, the device issues a new server certificate (that is a proxy server certificate) to the clients for SSL negotiation according to the received server certificate. Through the proxy server certificate, the internal clients can obtain the device identity and know whether the external server is trusted.

When the device proxies the SSL server to establish a connection with the client, it need to send a certificate to the client to identify itself. Instead of sending the certificate of the real server, the device uses the SSL decryption certificate to issue a new certificate (proxy server certificate) based on the real server certificate.

After receiving the certificate of the real server, the device verifies the legitimacy of the certificate on behalf of the SSL client.

- If the server certificate is legitimate, the device uses the trusted SSL decrypted certificate to issue the proxy server certificate. The client considers the proxy server certificate as legitimate because it trusts the SSL decryption certificate used to issue the certificate.
- If the server certificate is illegitimate, the device uses the untrusted SSL decryption certificate to issue the proxy server certificate. The client considers the proxy server certificate as illegitimate because it does not trust the SSL decryption certificate used to issue the certificate. A security alarm will be generated to indicate that the certificate is illegitimate and users must clear the alarm to continue the access.

Internal server certificate

The internal server certificates are required in the scenario of protecting internal servers. With an internal server certificate imported, the device will decrypt the certificate and generate a CER file and a key file.

- **CER file**—Used to identify the server. The device will calculate the MD5 value of the CER file and use the MD5 value as the unique identifier of the file. During the SSL proxy process, the device first calculates the MD5 value of the received server certificate and then compares the calculated MD5 value with the MD5 value of the imported internal server certificate. If they are the same, the certificate is trusted. If they are different, the certificate is untrusted.
- **Key file**—Used to encrypt and decrypt the packets in the subsequent SSL proxy process.

SSL proxy whitelist

The device does not proxy SSL connections destined for a server if the IP address or domain name of the server is on the SSL proxy whitelist.

The device supports the following types of SSL proxy whitelist lists:

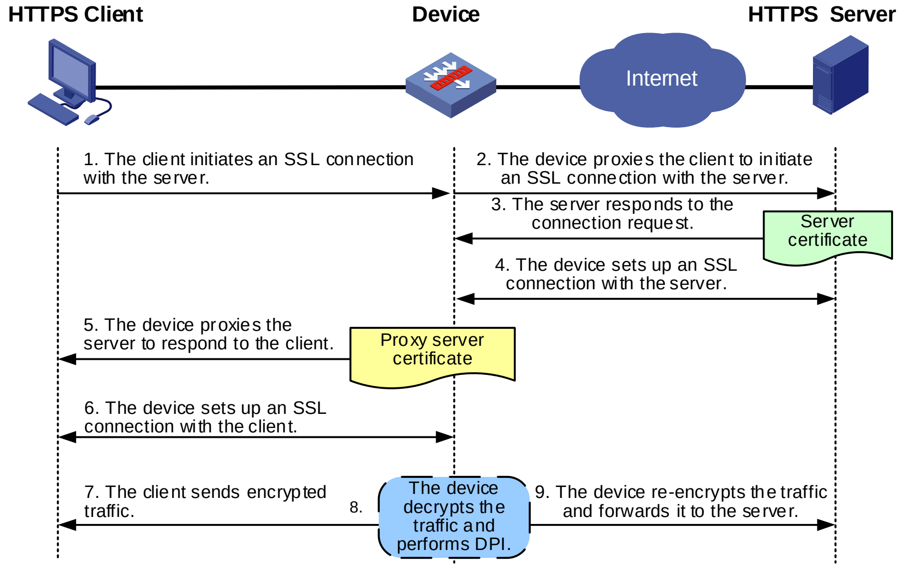
- **Predefined**—The device provides a predefined SSL hostname whitelist and a predefined SSL IP address whitelist.
- **User-defined**—The device supports a user-defined SSL hostname list. You can disable SSL proxy for SSL connections destined for a server by adding the server hostname to the whitelist.

Mechanism for protecting internal clients

The SSL proxy function is implemented based on the TCP proxy function. When the device detects SSL traffic that needs SSL proxy, it first acts as the TCP proxy to set up TCP connections, and then implements SSL proxy.

Figure 3 shows how the device acts as an SSL proxy to protect the internal client when it accesses the external server.

Figure 3 SSL proxy mechanism for protecting the internal client


HTTPS Client
Device
HTTPS Server
Internet
1. The client initiates an SSL connection with the server.
2. The device proxies the client to initiate an SSL connection with the server.
3. The server responds to the connection request.
Server certificate
4. The device sets up an SSL connection with the server.
5. The device proxies the server to respond to the client.
Proxy server certificate
6. The device sets up an SSL connection with the client.
7. The client sends encrypted traffic.
8.
The device decrypts the traffic and performs DPI.
9. The device re-encrypts the traffic and forwards it to the server.

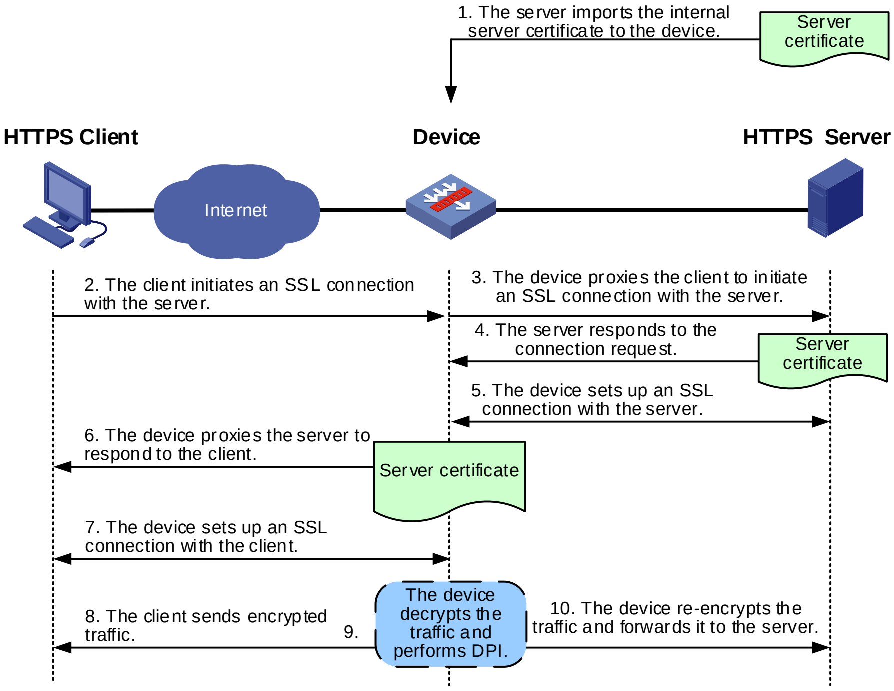
1. The client sends a request to initiate an SSL connection with the server.
2. The device receives the request from the client and proxies the client to send a new SSL connection request to the server.
3. The server responds to the request and sends the server certificate to the device.
4. The device verifies the legitimacy of the server certificate, completes the SSL handshake with the server on behalf of the SSL client, and sets up an SSL connection.
5. The device acts as the SSL proxy server to respond to the client connection request and send a proxy server certificate to the client. The proxy server certificate is issued by using the SSL decryption certificate based on the content of the real server certificate.
6. The client verifies the proxy server certificate, completes the SSL handshake with the device, and sets up an SSL connection.
7. The client and server transmit encrypted SSL traffic with the device, respectively.
8. The device decrypts the SSL traffic from the client or server and performs deep packet inspection and audit on the traffic.
9. The device re-encrypts the traffic and sends it to the server.

Mechanism for protecting internal servers

The SSL proxy function is implemented based on the TCP proxy function. When the device detects SSL traffic that needs SSL proxy, it first acts as the TCP proxy to set up TCP connections, and then implements SSL proxy.

Figure 3 shows how the device acts as an SSL proxy to protect the internal server when it is accessed by the external client.

Figure 4 SSL proxy mechanism for protecting the internal server


1. The server imports the internal server certificate to the device.
Server certificate
HTTPS Client
Device
HTTPS Server
Internet
2. The client initiates an SSL connection with the server.
3. The device proxies the client to initiate an SSL connection with the server.
4. The server responds to the connection request.
Server certificate
5. The device sets up an SSL connection with the server.
6. The device proxies the server to respond to the client.
Server certificate
7. The device sets up an SSL connection with the client.
8. The client sends encrypted traffic.
9.
The device decrypts the traffic and performs DPI.
10. The device re-encrypts the traffic and forwards it to the server.

1. The administrator obtains the certificate of the internal server to be protected and imports the internal server certificate to the device.
2. The device receives the SSL connection request from the client.
3. The device proxies the client to send a new SSL connection request to the server.
4. The server responds to the request and sends the server certificate to the device.
5. The device verifies the legitimacy of the server certificate, completes the SSL handshake with the server on behalf of the SSL client, and sets up an SSL connection.
6. The device acts as the SSL proxy server to respond to the client connection request and send the imported internal server certificate to the client.
7. The client verifies the internal server certificate, completes the SSL handshake with the device, and sets up an SSL connection.
8. The client and server transmit encrypted SSL traffic with the device, respectively.
9. The device decrypts the SSL traffic from the client or server and performs deep packet inspection and audit on the traffic.
10. The device re-encrypts the traffic and sends it to the server.

Proxy policy rules

The device supports only one proxy policy.

The proxy policy contains a set of user-defined proxy policy rules. Each rule defines a set of traffic filtering criteria and an action (TCP proxy, SSL decryption, or no-proxy) for traffic matching the rule.

Naming and numbering of proxy policy rules

Each proxy policy rule is identified by a unique rule name and ID. The rule name is required but the rule ID is optional when you create a rule. If you do not specify a rule ID, the system automatically assigns an ID to the rule.

Traffic filtering criteria in a proxy policy rule

You can configure the following types of criteria to filter the traffic to which a proxy policy rule applies:

- Source security zone.
- Destination security zone.
- Source IP address.
- Destinations IP address.
- User.
- User group.
- Service.

You can configure multiple criteria for each filtering criteria type. A packet matches a filtering criteria type if it matches a filtering criterion of the type. For example, you can configure multiple source security zone filtering criteria in a proxy policy rule. A packet passes the source security zone filtering if it matches any of the configured source security zone filtering criteria.

Proxy policy rule match order

By default, proxy policy rules are matched in the order they were created. A rule created earlier has a higher priority. You can change the rule match order by rearranging proxy policy rules. The match process stops once a matching rule is found.

The more refined the traffic filtering criteria are, the smaller the application range of the proxy policy rule. Configure the proxy policy rules in ascending order of their application ranges as a best practice.

Proxy policy rule actions

The device supports the following actions for traffic matching a proxy policy rule:

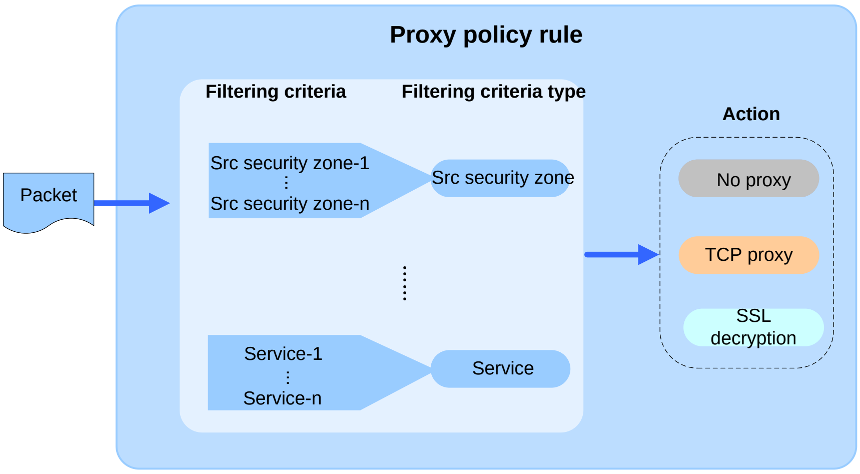
- **TCP-proxy**—The device acts as a TCP proxy and provides TCP-layer isolation between the TCP client and TCP server.
- **SSL-decryption**—The device acts as an SSL proxy to decrypt the SSL traffic between the SSL client and server and to implement deep packet inspection on the decrypted traffic.

  The SSL decryption supports the following protection services:

  - Internal client protection.
  - Internal server protection.
- **No-proxy**—The device directly transmits the traffic without TCP or SSL proxy.

Matching process

Figure 5 Matching process



As shown in Figure 5, a packet is compared against the proxy policy rules as follows:

1. The device compares the packet against the proxy policy rule with the highest priority.

The proxy policy rule can contain multiple filtering criteria types and each filtering criteria type can contain multiple filtering criteria. The packet matches a filtering criteria type if it matches any filtering criterion of the type.

2. The device determines that the packet matches the proxy policy rule if the packet matches each of following filtering criteria types in the rule:
   - Source security zone.
   - Destination security zone.
   - Source IP address.
   - Destinations IP address.
   - Service.
   - Either the user or user group filtering criteria type.
3. If the packet does not match the highest priority rule, the device continues the match process. The match process stops until a matching rule is found, and the action associated with the matching rule is applied.
4. If the packet does not match any rules in the proxy policy, the device applies the default action specified in the proxy policy to the packet.

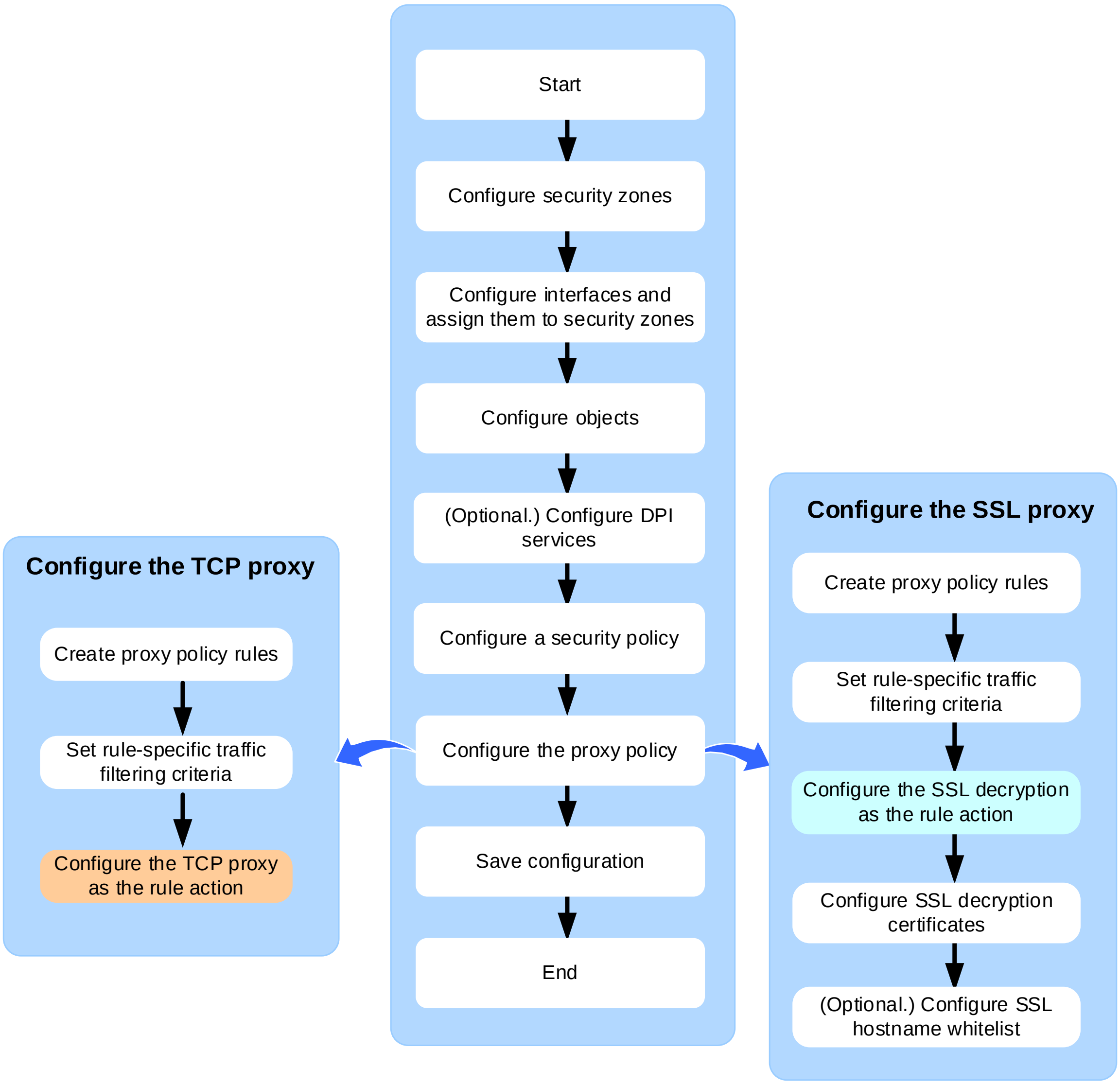
Restrictions and guidelines: Proxy policy configuration

- The TCP proxy and SSL proxy functions impact the device forwarding performance. When you configure proxy policy rules, refine the traffic filtering criteria to restrict application of the rules only to traffic that requires TCP or SSL proxy.
- Proxy policy rules are matched in descending order of their priorities. In the scenario that requires both TCP proxy and SSL decryption, make sure the proxy policy rule with the SSL decryption action has a higher priority than the rule with the TCP proxy action. This ensures that SSL traffic that requires SSL decryption can match the SSL decryption rule first.
- In the security policy to which the proxy policy is applied, make sure communications between the source security zones and the **Local** security zone are permitted. For more information about security zones, see security zone configuration in *Security Configuration Guide*.
- After the SSL proxy function is enabled, the packet capture action of the intrusion prevention system will be invalid. For more information about IPS, see "Configuring IPS."
- Select a protection service of the SSL decryption as required and import the corresponding certificates to the device for SSL connection establishment with the clients.
- In a hot backup system that performs asymmetric forwarding, TCP proxy and SSL proxy are not supported. If you configure TCP proxy or SSL proxy, the configuration does not take effect. For more information about hot backup, see RBM-based hot backup configuration in *High Availability Configuration Guide*.

Proxy policy configuration procedure summary

Figure 6 shows how to configure the proxy policy.

Figure 6 Proxy policy configuration procedure



Start
Configure security zones
Configure interfaces and assign them to security zones
Configure objects
(Optional.) Configure DPI services
Configure a security policy
Configure the proxy policy
Save configuration
End
Configure the TCP proxy
Create proxy policy rules
Set rule-specific traffic filtering criteria
Configure the TCP proxy as the rule action
Configure the SSL proxy
Create proxy policy rules
Set rule-specific traffic filtering criteria
Configure the SSL decryption as the rule action
Configure SSL decryption certificates
(Optional.) Configure SSL hostname whitelist

Proxy policy tasks at a glance

To configure the proxy policy:

1. Configuring attributes in the proxy policy
   a. Setting the default action for the proxy policy
   b. Creating a proxy policy rule
   c. Configuring the traffic filtering criteria for a proxy policy rule
   d. Setting the action for a proxy policy rule
2. (Optional.) Managing proxy policy rules
   a. Changing the rule match order
   b. Disabling a proxy policy rule
3. Configuring SSL decryption certificates

   The SSL decryption certifications are required in the scenario of protecting internal clients.

   a. Importing an SSL decryption certificate

   b. (Optional.) Channing the credibility of an SSL decryption certificate
   c. (Optional.) Deleting an SSL decryption certificate
4. Configuring internal server certificates

   The internal server certificates are required in the scenario of protecting internal servers.

   a. Importing an internal server certificate
   b. (Optional.) Deleting an internal server certificate
5. (Optional.) Configuring the SSL proxy whitelist
   a. Add a hostname to the user-defined SSL hostname whitelist
   b. Disabling hostnames on the predefined SSL hostname whitelist
   c. Activating SSL hostname whitelist settings

Prerequisites for proxy policy configuration

Before you configure the proxy policy, complete the following tasks:

- Configure IP address object groups and service object groups. For more information, see object group configuration in *Security Configuration Guide*.
- Configure user and user groups. For more information, see user identification configuration in *Security Configuration Guide*.
- Configure security zones. For more information, see security zone configuration in *Security Configuration Guide*.
- Configure security policies. For more information, see security policy configuration in *Security Configuration Guide*.
- Configure DPI services. For more information, see *DPI Configuration Guide.*

Configuring attributes in the proxy policy

Setting the default action for the proxy policy

About this task

If a packet does not match any rules in the proxy policy or if the proxy policy does not contain any rules, the device applies the default action to the packet.

Procedure

1. Enter system view.

   `system-view`

2. Enter proxy policy view.

   `app-proxy-policy`

3. Set the default action for the proxy policy.

   `default action` { **`no-proxy`** | **`ssl-decrypt`** | **`tcp-proxy`** }

   By default, the device transmits packets that do not match any proxy policy rules directly without TCP or SSL proxy.

4. Specify an SSL decryption protection mode for the proxy policy.

   `default ssl-decrypt protect-mode` { **`client`** | **`server`** }

   By default, the SSL decryption protection mode of the proxy policy is **`client`**.

Creating a proxy policy rule

1. Enter system view.

   system-view

2. Enter proxy policy view.

   app-proxy-policy

3. Create a proxy policy rule and enter its view.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

Configuring the traffic filtering criteria for a proxy policy rule

Restrictions and guidelines

A proxy policy rule that does not contain any filtering criteria matches all packets.

A filtering criterion is ignored during packet matching if it is associated with a nonexistent or empty object group.

Procedure

1. Enter system view.

   system-view

2. Enter proxy policy view.

   app-proxy-policy

3. Enter the view of a proxy policy rule.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Configure the traffic filtering criteria as needed:
   - Configure a source security zone filtering criterion.

     source-zone *source-zone-name*

     By default, a proxy policy rule does not contain any source security zone filtering criterion.
   - Configure a destination security zone filtering criterion.

     destination-zone *destination-zone-name*

     By default, a proxy policy rule does not contain any destination security zone filtering criterion.
   - Configure a source IP address filtering criterion.

     source-ip object-group *object-group-name*

     By default, a proxy policy rule does not contain any source IP address filtering criterion.
   - Configure a destination IP address filtering criterion.

     destination-ip object-group *object-group-name*

     By default, a proxy policy rule does not contain any destination IP address filtering criterion.
   - Configure a service filtering criterion.

     service object-group { *object-group-name* | **any** }

     By default, a proxy policy rule does not contain any service filtering criterion.
   - Configure a user filtering criterion.

     user *username* [**domain** *domain-name*]

     By default, a proxy policy rule does not contain any user filtering criterion.
   - Configure a user group filtering criterion.

     user-group *user-group-name* [**domain** *domain-name*]

By default, a proxy policy rule does not contain any user group filtering criterion.

Setting the action for a proxy policy rule

1. Enter system view.

   `system-view`
2. Enter proxy policy view.

   `app-proxy-policy`
3. Enter the view of a proxy policy rule.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }
4. Set the action for traffic matching the proxy policy rule.

   action { **no-proxy** | **ssl-decrypt** | **tcp-proxy** }

   By default, the device transmits packets that match a proxy policy rule directly without TCP or SSL proxy.
5. Specify an SSL decryption protection mode for the proxy policy rule.

   ssl-decrypt protect-mode { **client** | **server** }

   By default, the SSL decryption protection mode of a proxy policy rule is **client**.

   Configure this command only when the SSL decryption action is used as the default action for the proxy policy.

Managing proxy policy rules

Changing the rule match order

About this task

By default, proxy policy rules are matched in the order they were created. A rule created earlier has a higher priority. You can change the rule match order by rearranging proxy policy rules. The match process stops once a matching rule is found.

Procedure

1. Enter system view.

   `system-view`
2. Enter proxy policy view.

   `app-proxy-policy`
3. Move a proxy policy rule.
   - Move a proxy policy rule through rule IDs.

     rule move id *rule-id* **before** *insert-rule-id*
   - Move a proxy policy rule through rule names.

     rule move name *rule-name1* { **before** [*rule-name2*] | **after** *rule-name2* }

Disabling a proxy policy rule

1. Enter system view.

   `system-view`
2. Enter proxy policy view.

app-proxy-policy

3. Enter the view of a proxy policy rule.

   rule { *rule-id* | [*rule-id*] **name** *rule-name* }

4. Disable the proxy policy rule.

   disable

   By default, a proxy policy rule is enabled.

Configuring SSL decryption certificates

Importing an SSL decryption certificate

About this task

The SSL decryption certifications are required in the scenario of protecting internal clients.

Restrictions and guidelines

The device supports a maximum of one trusted SSL decryption certificate and one untrusted SSL decryption certificate. Both of them must be manually imported to the device.

If you import multiple trusted or multiple untrusted SSL decryption certificates to the device, the most recent configuration takes effect.

You must install the trusted SSL decryption certificate on the client browser so the client can trust proxy server certificates issued by using the trusted SSL decryption certificate.

After an SSL decryption certificate is imported, its file extension will be changed to .cer.

Procedure

1. Enter system view.

   system-view

2. Import a CA certificate as a trusted or untrusted SSL decryption certificate to the device.

   app-proxy ssl-decrypt-certificate import { **trusted** | **untrusted** } { **pem** | **p12** } **filename** *filename*

Channing the credibility of an SSL decryption certificate

Restrictions and guidelines

The device supports only one trusted SSL decryption certificate and one untrusted SSL decryption certificate. A newly imported trusted or untrusted SSL decryption certificate will overwrite the existing one.

After an SSL decryption certificate is imported, its file extension will be changed to .cer. Append the .cer file extension when you specify the file containing the certificate whose credibility you want to change.

Procedure

1. Enter system view.

   system-view

2. Change the credibility of an SSL decryption certificate.

   app-proxy ssl-decrypt-certificate modify { **trusted** | **untrusted** } **filename** *filename*

Deleting an SSL decryption certificate

Restrictions and guidelines

The device, acting as an SSL proxy, requires the correct SSL decryption certificate to issue proxy server certificates to send to clients for server authentication. If the required SSL decryption certificate is not available, the device cannot set up a connection with the client and the SSL traffic will be transmitted directly without SSL decryption.

After an SSL decryption certificate is imported, its file extension will be changed to .cer, which must be appended to the file name when you delete the certificate.

Procedure

1. Enter system view.
   system-view
2. Delete an SSL decryption certificate.
   app-proxy ssl-decrypt-certificate delete filename *filename*

Configuring internal server certificates

Importing an internal server certificate

About this task

The internal server certifications are required in the scenario of protecting internal servers.

Restrictions and guidelines

Each internal server certificate stored in the device has a MD5 value. If the MD5 value of an internal server certificate to be imported is the same as that stored in the device, the new certificate will overwrite the old certificate.

Procedure

1. Enter system view.
   system-view
2. Import an internal server certificate.
   app-proxy internal-server-certificate import { **p12** | **pem** } **filename** *filename*

Deleting an internal server certificate

About this task

When an internal server certificate expires or an internal server does not need to be protected, you can execute this command to delete the imported internal server certificate.

You can execute the **display app-proxy imported internal-server-certificate** command to view the MD5 values of the internal server certificates.

Procedure

1. Enter system view.
   system-view
2. Delete an internal server certificate.
   app-proxy internal-server-certificate delete md5 *md5-value*

Configuring the SSL proxy whitelist

Add a hostname to the user-defined SSL hostname whitelist

About this task

In the scenario that requires SSL client authentication or in-depth server certificate inspection, SSL proxy will cause the device (as an SSL proxy) unable to pass SSL client or server verification. By adding the server's hostname to the SSL hostname whitelist, you can disable SSL proxy for connections destined for the server so the SSL traffic is transmitted directly without SSL decryption.

Procedure

1. Enter system view.

   system-view

2. Add the hostname of a server to the user-defined SSL hostname whitelist.

   app-proxy ssl whitelist user-defined-hostname *host-name*

   By default, the user-defined SSL hostname whitelist does not contain any hostnames.

Disabling hostnames on the predefined SSL hostname whitelist

1. Enter system view.

   system-view

2. Disable hostnames on the predefined SSL hostname whitelist.

   undo app-proxy ssl whitelist predefined-hostname { **chrome-hsts** [*hostname*] | *hostname* } **enable**

   By default, the entire predefined SSL hostname whitelist is enabled.

Activating SSL hostname whitelist settings

About this task

You must manually activate the SSL hostname whitelist settings for the following configurations to take effect:

- Adding or removing hostnames to or from the user-defined SSL hostname whitelist.
- Enabling or disabling hostnames on the predefined SSL hostname whitelist.

Procedure

1. Enter system view.

   system-view

2. Activate SSL hostname whitelist configuration.

   app-proxy ssl whitelist activate

Display and maintenance commands for the proxy policy

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display proxy policy information.	**display app-proxy-policy**
Display information about imported internal server certificates.	**display app-proxy imported internal-server-certificate**
Display SSL decryption certificate information.	**display app-proxy ssl-decrypt-certificate**
Display the SSL server certificates received by the device as the SSL proxy client.	**display app-proxy server-certificate** [**slot** *slot-number*]
Display the predefined or user-defined SSL hostname whitelist.	**display app-proxy ssl whitelist hostname** { **user-defined** | **predefined** }
Display IP addresses on the SSL IP address whitelist.	**display app-proxy ssl whitelist** { **ipv4** | **ipv6** } { **all** [slot *slot-number*] | *ip-address* }
Clear information about the SSL server certificates received by the device as the SSL proxy client.	**reset app-proxy server-certificate**
Clear the SSL IP address whitelist.	**reset app-proxy ssl whitelist ip**

Proxy policy configuration examples

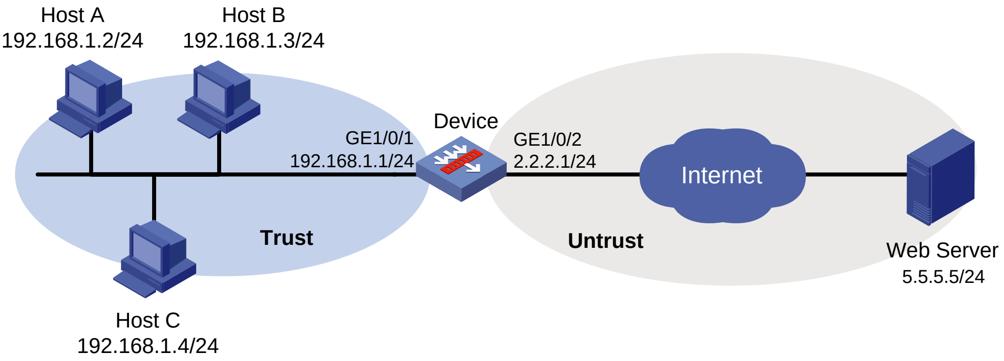
Example: Configuring the proxy policy

Network configuration

As shown in Figure 7, the device connects the LAN and the Internet. The LAN resides in security zone **Trust** and the Internet resides in security zone **Untrust**.

Configure the device to decrypt the HTTPS traffic and perform deep packet inspection on the traffic.

Figure 7 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
```

```
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [Device] ip route-static 5.5.5.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Create an IP address object group named **obj1** and configure an IP address object with subnet **192.168.1.0/24**.

   ```
   [Device] object-group ip address obj1
   [Device-obj-grp-ip-obj1] network subnet 192.168.1.0 24
   [Device-obj-grp-ip-obj1] quit
   ```

5. Import CA certificate **trust.pem** as a trusted SSL decryption certificate and CA certificate **untrust.pem** as an untrusted SSL decryption certificate to the device.

   ```
   [Device] app-proxy ssl-decrypt-certificate import trust pem filename trust.pem
   [Device] app-proxy ssl-decrypt-certificate import untrust pem filename untrust.pem
   ```

6. Install CA certificate **trust.pem** on all internal hosts and set the CA certificate **trust.pem** as a trusted SSL decryption certificate. (Details not shown.)

7. Create proxy policy rule **https** to match HTTPS traffic from internal hosts destined for the Web server and set the action for traffic matching rule **https** to SSL decryption.

   ```
   [Device] app-proxy-policy
   [Device-app-proxy-policy] rule 1 name https
   [Device-app-proxy-policy-rule-1-https] source-zone trust
   [Device-app-proxy-policy-rule-1-https] destination-zone untrust
   [Device-app-proxy-policy-rule-1-https] source-ip object-group obj1
   [Device-app-proxy-policy-rule-1-https] service object-group https
   [Device-app-proxy-policy-rule-1-https] action ssl-decrypt
   [Device-app-proxy-policy-rule-1-https] quit
   ```

8. Configure URL filtering:

   # Create user-defined URL category **https**, set its severity level to **1001**, and create URL filtering rule 1 to match HTTPS packets that contain host name **www. baidu.com** in the URL.

   ```
   [Device] url-filter category https severity 1001
   [Device-url-filter-category-https] rule host text www.baidu.com
   [Device-url-filter-category-https] quit
   ```

   # Create a URL filtering policy named **p1**. Specify action **drop** for URL category **https**, enable logging for the matching packets.

   ```
   [Device] url-filter policy p1
   [Device-url-filter-policy-p1] category https action reset logging
   [Device-url-filter-policy-p1] quit
   ```

9. Apply URL filtering policy **p1** to a DPI application profile and activate the URL filtering policy settings:

# Create a DPI application profile named **sec**, and apply URL filtering policy **p1** to the DPI application profile.

```
[Device] app-profile sec
[Device-app-profile-sec] url-filter apply policy p1
[Device-app-profile-sec] quit
```

# Activate the URL filtering policy and rule settings.

```
[Device] inspect activate
```

10. Configure a security policy:

# Create a security policy rule named **trust-untrust** to permit the packets from security zone **Trust** to security zone **Untrust** and to perform URL filtering on the packets. Make you have configured IPS on the device.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-10-trust-untrust] source-zone trust
[Device-security-policy-ip-10-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-10-trust-untrust] destination-zone untrust
[Device-security-policy-ip-10-trust-untrust] action pass
[Device-security-policy-ip-10-trust-untrust] profile sec
[Device-security-policy-ip-10-trust-untrust] quit
```

# Create a security policy rule named **untrust-trust to permit the packets from** security zone **Untrust** to security zone **Trust.**

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-11-untrust-trust] source-zone untrust
[Device-security-policy-ip-11-untrust-trust] destination-zone trust
[Device-security-policy-ip-11-untrust-trust] destination-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-11-untrust-trust] action pass
[Device-security-policy-ip-11-untrust-trust] profile sec
[Device-security-policy-ip-11-untrust-trust] quit
```

# Create two security policy rules named **proxyserverlocalin and proxyserverlocalout to** permit the traffic between the **Trust** and **Local** security zones**. Thus, the device can proxy the traffic** from internal hosts destined for the Web server**.**

```
[Device-security-policy-ip] rule name proxyserverlocalin
[Device-security-policy-ip-12-proxyserverlocalin] source-zone trust
[Device-security-policy-ip-12-proxyserverlocalin] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-12-proxyserverlocalin] destination-zone local
[Device-security-policy-ip-12-proxyserverlocalin] action pass
[Device-security-policy-ip-12-proxyserverlocalin] quit
[Device-security-policy-ip] rule name proxyserverlocalout
[Device-security-policy-ip-13-proxyserverlocalout] source-zone local
[Device-security-policy-ip-13-proxyserverlocalout] destination-zone trust
[Device-security-policy-ip-13-proxyserverlocalout] destination-ip-subnet
192.168.1.0 24
[Device-security-policy-ip-13-proxyserverlocalout] action pass
[Device-security-policy-ip-13-proxyserverlocalout] quit
```

# Create two security policy rules named **proxyclientlocalin and proxyclientlocalout to** permit the traffic between the **Untrust** and **Local** security zones**. Thus, the device can proxy the traffic** from the Web server destined for internal hosts**.**

```
[Device-security-policy-ip] rule name proxyclientlocalin
```

```
[Device-security-policy-ip-14-proxyclientlocalin] source-zone untrust
[Device-security-policy-ip-14-proxyclientlocalin] destination-zone local
[Device-security-policy-ip-14-proxyclientlocalin] destination-ip-subnet
192.168.1.0 24
[Device-security-policy-ip-14-proxyclientlocalin] action pass
[Device-security-policy-ip-14-proxyclientlocalin] quit
[Device-security-policy-ip] rule name proxyclientlocalout
[Device-security-policy-ip-15-proxyclientlocalout] source-zone local
[Device-security-policy-ip-15-proxyclientlocalout] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-15-proxyclientlocalout] destination-zone untrust
[Device-security-policy-ip-15-proxyclientlocalout] action pass
[Device-security-policy-ip-15-proxyclientlocalout] quit
```

# Activate rule matching acceleration.

```
[Device-security-policy-ip] accelerate enhanced enable
[Device-security-policy-ip] quit
```

Verifying the configuration

Verify that the device can deny HTTPS requests to www.baidu.com and generate logs. You can also execute the **display app-proxy server-certificate** command to view the following information:

- SSL server certificates received by the device as the SSL proxy client.
- Number of times connections to the server had been proxied.
- Most recent time the device proxied a connection to the server.
- First time the device proxied a connection to the server.

NSFOCUS Firewall Series

NF NAT Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for NAT, NAT66, and AFT.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

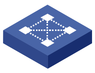
GUI conventions

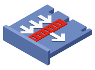
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

NAT overview

Network Address Translation (NAT) translates an IP address in the IP packet header to another IP address. Typically, NAT is configured on gateways to enable private hosts to access external networks and external hosts to access private network resources such as a Web server.

Restrictions and guidelines: NAT configuration

The global NAT policy has priority over interface-based NAT. If both are configured, a packet is translated based on the global NAT policy when the packet matches an ACL rule in the global NAT policy. The interface-based source and destination address translation rules do not take effect. As a best practice, do not configure both the global NAT and interface-based NAT policies.

A device does not perform AFT translation on NATed packets.

Basic NAT concepts

The following describes basic NAT concepts:

- **NAT device**—A device configured with NAT. Typically, NAT is configured on the edge device that connects the internal and external networks.
- **NAT interface**—An interface configured with NAT.
- **NAT rule**—Rules that define how to perform address translation.
- **NAT address**—A public IP address used for address translation, and this address is reachable from the external network. The NAT address can be manually assigned or dynamically obtained.
- **NAT entry**—Stores the mapping between a private IP address and a public IP address. For more information, see "NAT entries."
- **Easy IP**—Uses the IP address of an interface as the NAT address. The IP address of the interface can be manually assigned or be obtained through DHCP.
- **Global NAT**—Uses NAT rules configured for the global NAT policy to translate packets.
- **Interface-based NAT**—Uses NAT rules configured on a per-interface basis to translate packets.

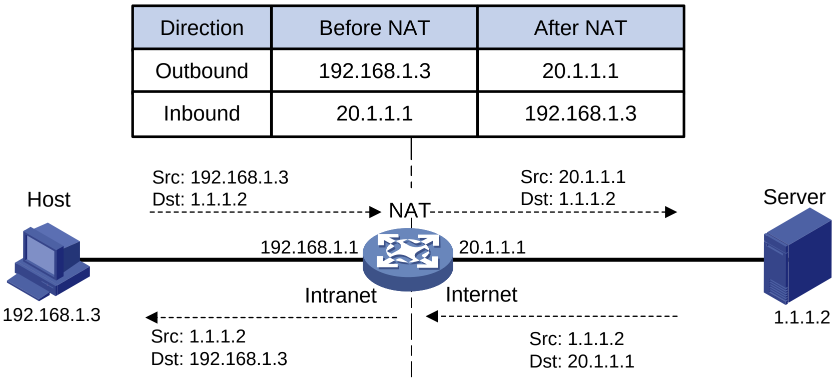
Basic NAT operating mechanism

Figure 1 shows the basic NAT operating mechanism.

1. Upon receiving a request from the host to the server, NAT translates the private source address 192.168.1.3 to the public address 20.1.1.1 and forwards the NATed packet. NAT adds a mapping for the two addresses to its NAT table.
2. Upon receiving a response from the server, NAT translates the destination public address to the private address, and forwards the packet to the host.

The NAT operation is transparent to the terminals (the host and the server). NAT hides the private network from the external users and shows that the IP address of the internal host is 20.1.1.1.

Figure 1 Basic NAT operation



Direction	Before NAT	After NAT
Outbound	192.168.1.3	20.1.1.1
Inbound	20.1.1.1	192.168.1.3

NAT control

You can use ACLs to implement NAT control. The match criteria in the ACLs include the source IP address, source port number, destination IP address, destination port number, transport layer protocol, and VPN instance. Only packets permitted by an ACL are processed by NAT.

NAT translation methods

Static NAT

Static NAT creates a fixed mapping between a private address and a public address. It supports connections initiated from internal users to external network and from external users to the internal network. Static NAT applies to regular communications.

Source address translation

Source address translation is dynamic NAT translation that uses an address pool to translate addresses. It applies to the scenario where a large number of internal users access the external network.

The NO-PAT, port-based PAT, and port block-based PAT modes are supported.

NO-PAT

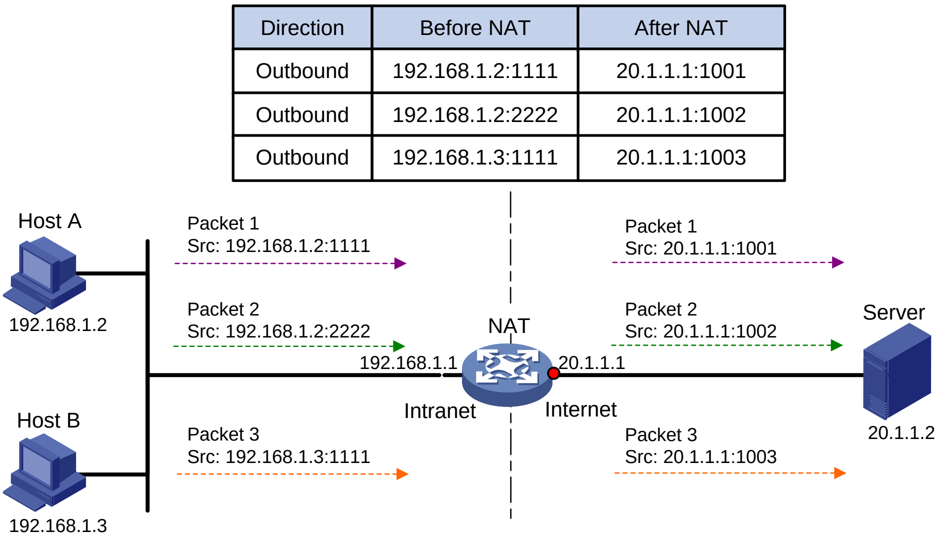
Not Port Address Translation (NO-PAT) translates a private IP address to an IP public address. The public IP address cannot be used by another internal host until it is released.

NO-PAT supports all IP packets.

Port-based PAT

Port Address Translation (PAT) translates multiple private IP addresses to a single public IP address by mapping the private IP address and source port to the public IP address and a unique port. PAT supports TCP and UDP packets, and ICMP request packets.

Figure 2 PAT operation



Direction	Before NAT	After NAT
Outbound	192.168.1.2:1111	20.1.1.1:1001
Outbound	192.168.1.2:2222	20.1.1.1:1002
Outbound	192.168.1.3:1111	20.1.1.1:1003

As shown in Figure 2, PAT translates the source IP addresses of the three packets to the same IP public address and translates their port numbers to different port numbers. Upon receiving a response, PAT translates the destination address and port number of the response, and forwards it to the target host.

PAT supports the following mappings:

- **Endpoint-Independent Mapping (EIM)**—Uses the same IP and port mapping (EIM entry) for packets from the same source IP and port to any destinations. EIM allows external hosts to initiate connections to the translated IP addresses and ports of internal hosts. It allows internal hosts behind different NAT gateways to access each other.
- **Address and Port-Dependent Mapping (APDM)**—Uses different IP and port mappings for packets from the same source IP and port to different destination IP addresses and ports. APDM allows an external host to initiate connections to an internal host only under the condition that the internal host has previously accessed the external host. It is secure, but it does not allow internal hosts behind different NAT gateways to access each other.

Port block-based NAT

Port block-based NAT is a PAT translation based on port ranges. It maps multiple private IP addresses to one public IP address and uses a different port block for each private IP address. For example, the private IP address 10.1.1.1 of an internal host is mapped to the public IP address 202.1.1.1 and port block 10001 to 10256. When the internal host accesses public hosts, the source IP address 10.1.1.1 is translated to 202.1.1.1, and the source ports are translated to ports in the port block 10001 to 10256.

Port block-based NAT includes static and dynamic mappings. It applies to NAT444 and DS-Lite networks.

Static port block mapping

The NAT gateway computes a static port block mapping before address translation. The mapping is between a private IP address and a public IP address with a port block.

When an internal user initiates a connection to the external network, the system performs the following operations:

- Locates a static mapping based on the private IP address of the user and obtains the public IP address and the port block in the mapping.
- Selects a public port number in the port block.

- Translates the private IP address to the public IP address and assigns the selected public port number.

The NAT gateway uses private IP addresses, public IP addresses, a port range, and a port block size to compute static mappings:

1. Divides the port range by the port block size to get the number of available port blocks for each public IP address.

   This value is the base number for mapping.
2. Sorts the port blocks in ascending order of the start port number in each block.
3. Sorts the private IP addresses and the public IP addresses separately in ascending order.
4. Maps the first base number of private IP addresses to the first public IP address and its port blocks in ascending order.

For example, the number of available port blocks of each public IP address is **m**. The first **m** private IP addresses are mapped to the first public IP address and the **m** port blocks in ascending order. The next **m** private IP addresses are mapped to the second IP address and the **m** port blocks in ascending order. The other static port block mappings are created by analogy.

Dynamic port block mapping

When an internal user initiates a connection to the external network, the dynamic port block-based NAT operates as follows:

1. Uses ACLs to implement translation control. It processes only packets that match an ACL permit rule.
2. Creates a mapping from the internal user's private IP address to a public IP address and a port block.
3. Translates the private IP address to the public IP address, and the source ports to ports in the selected port block for subsequent connections from the private IP address.
4. Withdraws the port block and deletes the dynamic port block mapping when all connections from the private IP address are disconnected.

Dynamic port block mapping supports port block extending. If the ports in the port block for a private address are all occupied, dynamic port block mapping translates the source port to a port in an extended port block.

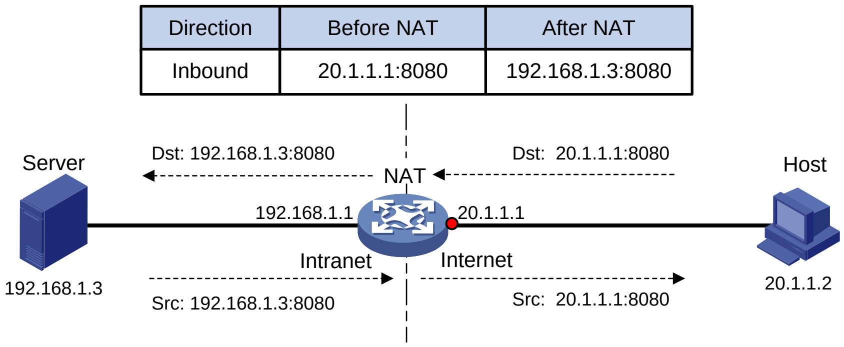
Destination address translation

Destination address translation maps a public address and port number to the private IP address and port number of an internal server. This feature allows servers in the private network to provide services for external users.

Figure 3 shows how destination address translation works:

1. Upon receiving a request from the host, NAT translates the public destination IP address and port number to the private IP address and port number of the internal server.
2. Upon receiving a response from the server, NAT translates the private source IP address and port number to the public IP address and port number.

Figure 3 Destination address translation operation



Direction	Before NAT	After NAT
Inbound	20.1.1.1:8080	192.168.1.3:8080

NAT entries

NAT session entry

NAT creates a NAT session entry for a session and creates an address mapping for the first packet in the session.

A NAT session entry contains extended NAT information, such as interface and translation method. Subsequent packets of the session are translated by using this entry.

- If the direction of the subsequent packets is the same as the direction of the first translated packet, NAT performs the source and destination address translation the same as the first packet.
- If the direction of the subsequent packets is opposite to the direction of the first translated packet, NAT perform reverse address translation. For example, if the source address of the first packet is translated, then the destination address of the subsequent packets is translated.

The session management module maintains the updating and aging of NAT session entries. For information about session management, see *Security Configuration Guide*.

EIM entry

If EIM is configured on the NAT device, the PAT mode will first create a NAT session entry, and then an EIM entry. The EIM entry is a 3-tuple entry, and it maps a private address/port to a public address/port. The EIM entry ensures:

- Subsequent new connections originating from the same source IP and port uses the same translation as the initial connection.
- Translates the address for new connections initiated from external hosts to the NAT address and port number based on the EIM entry.

An EIM entry ages out after all related NAT session entries age out.

NO-PAT entry

A NO-PAT entry maps a private address to a public address. The same mapping applies to subsequent connections originating from the same source IP.

A NO-PAT entry can also be created during the ALG process for NAT. For information about NAT ALG, see "NAT ALG."

A NO-PAT entry ages out after all related NAT session entries age out.

Port block-based entry

A port block-based entry maps a private IP address to a public IP address and a port block.

Port block-based entries include static and dynamic port block mappings. For information about these mappings, see "Static port block mapping" and "Dynamic port block mapping."

VRF-aware NAT

VRF-aware NAT allows users from different VRF (VPN instances) to access external networks and to access each other.

1. Upon receiving a request from a user in a VRF to an external network, NAT performs the following tasks:
   - Translates the private source IP address and port number to a public IP address and port number.
   - Records the VRF information, such as the VRF name.
2. When a response packet arrives, NAT performs the following tasks:
   - Translates the destination public IP address and port number to the private IP address and port number.
   - Forwards the packet to the target VRF.

The NAT Server feature supports VRF-aware NAT for external users to access the servers in a VPN instance. For example, to enable a host at 10.110.1.1 in VPN 1 to provide Web services for Internet users, configure NAT Server to use 202.110.10.20 as the public IP address of the Web server.

VRF-aware NAT is supported only in global NAT in the current software version.

NAT hairpin

NAT hairpin allows internal hosts to access each other through NAT. The source and destination IP address of the packets are translated on the interface connected to the internal network.

NAT hairpin includes P2P and C/S modes:

- **P2P**—Allows internal hosts to access each other through NAT. The internal hosts first register their public addresses to an external server. Then, the hosts communicate with each other by using the registered IP addresses.
- **C/S**—Allows internal hosts to access internal servers through NAT addresses. The destination IP address of the packet going to the internal server is translated by matching the NAT Server configuration. The source IP address is translated by matching the outbound dynamic or static NAT entries.

NAT ALG

NAT ALG (Application Level Gateway) translates address or port information in the application layer payloads to ensure connection establishment.

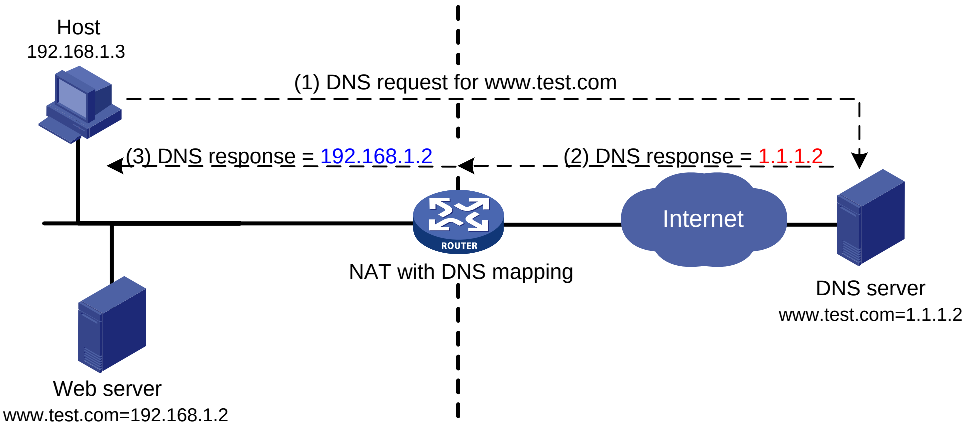
For example, an FTP application includes a data connection and a control connection. The IP address and port number for the data connection depend on the payload information of the control connection. This requires NAT ALG to translate the address and port information for data connection establishment.

NAT DNS mapping

The DNS server is typically on the public network. For the users on the public network to access an internal server, you can configure the NAT Server feature on the NAT device. The NAT Server maps the public IP address and port number to the private IP address and port number of the internal server. Then the public users can access the internal server through the server's domain name or public IP address.

When a user is in the private network, the user cannot access the internal server by using the domain name of the server. This is because the DNS response contains the public IP address of the server. In this case, you can configure NAT DNS mapping to solve the problem.

Figure 4 NAT DNS mapping



As shown in Figure 4, NAT DNS mapping works as follows:

1. The host sends a DNS request containing the domain name of the internal Web server.
2. Upon receiving the DNS response, the NAT device performs a DNS mapping lookup by using the domain name in the response. A NAT DNS mapping maps the domain name to the public IP address, public port number, and the protocol type for the internal server.
3. If a match is found, the NAT continues to compare the public address, public port number, and the protocol type with the NAT Server configuration. The NAT Server configuration maps the public IP address and port number to the private IP address and port number for the internal server.
4. If a match is found, NAT translates the public IP address in the response into the private IP address of the Web server.
5. The internal host receives the DNS response, and obtains the private IP address of the Web server.

NAT444

About NAT444

NAT444 provides carrier-grade NAT by unifying the NAT444 gateway, AAA server, and log server. NAT444 introduces a second layer of NAT on the carrier side, with few changes on the customer side and the application server side. With port block assignment, NAT444 supports user tracking. It has become a preferred solution for carriers in transition to IPv6.

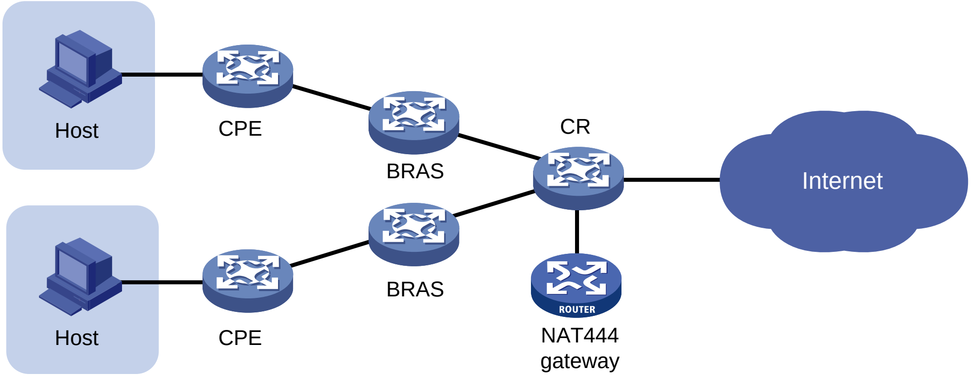
Centralized NAT444 deployment

Centralized NAT444 deployment is implemented by installing a NAT processing slot on the CR device or by connecting a NAT444 device to the CR.

As shown in Figure 5, when an internal user accesses the external network, NAT444 is implemented as follows:

1. The CPE device performs the first NAT.
2. After the user passes AAA authentication on the BRAS device, this user is assigned a private IP address.
3. When the packet destined to the external network, the NAT444 gateway performs the second NAT.

Figure 5 Centralized NAT444 deployment



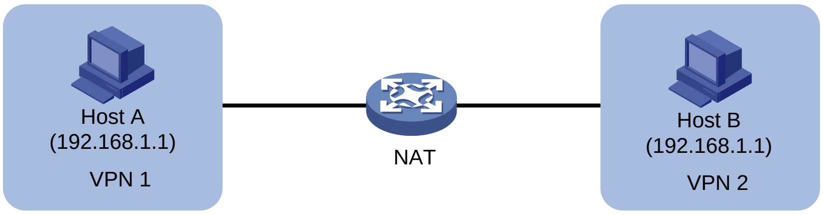
NAT for overlapping addresses

NAT for overlapping addresses in different VPNs

As shown in Figure 6, two hosts are in different VPN instances with overlapping addresses. For the hosts to access each other, both the source and destination addresses of packets between the two VPNs need to be translated. Configure static NAT on both interfaces connected to the VPNs on the NAT device.

1. Configure a static outbound NAT mapping between 192.168.1.1 in VPN 1 and 172.16.1.1 in VPN 2.
2. Configure a static outbound NAT mapping between 192.168.1.1 in VPN 2 and 172.16.2.1 in VPN 1.
3. When static NAT takes effect, the hosts can access each other.

Figure 6 VPN access with overlapping address

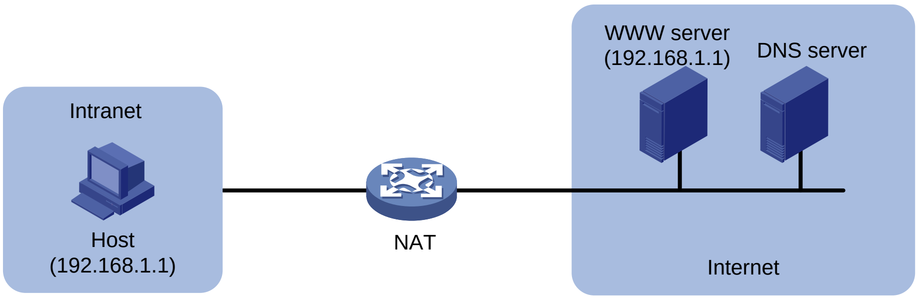


NAT for internal-to-external access through domain name

As shown in Figure 7, the IP address of the Web server overlaps with the private host at 192.168.1.0/24. Configure dynamic NAT ALG and outbound dynamic NAT to allow the internal host to access the external Web server by using the server's domain name.

1. The host sends a DNS request to the DNS server in the external network.
2. After receiving a DNS reply, the NAT device with NAT ALG configured translates the Web server's IP address in the DNS reply payload to a dynamically assigned public address 10.1.1.1.
3. Configure inbound dynamic NAT ALG to make sure the internal host reaches the Web server instead of another internal host. NAT ALG can translate the Web server's IP address in the DNS reply payload to a dynamically assigned public address 10.1.1.1.
4. After receiving the DNS reply from the NAT device, the host sends a packet with the source IP address 192.168.1.1 and destination IP address 10.1.1.1.
5. The NAT device with outbound dynamic NAT configured translates the source IP address of the packet to a dynamically assigned public address 20.1.1.1. NAT ALG translates the destination IP address of the packet to the IP address of the Web server.

Figure 7 Internal-to-external access through domain name



NAT in the DS-Lite network

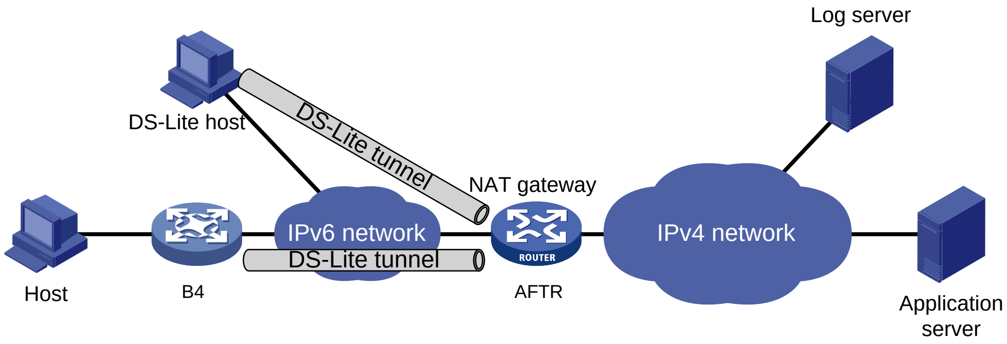
DS-Lite combines tunneling and NAT to allow an IPv4 private network to access the IPv4 public network over an IPv6 network. For more information about DS-Lite, see tunneling configuration in *VPN Configuration Guide*.

DS-Lite B4 address translation is configured on the AFTR and performs port block-based translation based on the IPv6 address of the B4 element. The B4 element refers to a B4 router or a DS-Lite host. DS-Lite B4 address translation dynamically maps a public IPv4 address and a port block to the IPv6 address of the B4 element. The DS-Lite host or hosts behind the B4 router use the mapped public IPv4 address and port block to access the public IPv4 network.

DS-Lite B4 address translation supports user tracing for DS-Lite hosts based on the port block.

Only dynamic port block mapping is supported for B4 address translation.

Figure 8 DS-Lite B4 address translation



Configuring global NAT

About the global NAT policy

The global NAT policy is applicable to scenarios where the external interface is not fixed. Compared with interface-based NAT policies, you do not need to change relevant configurations if the external interface changes, which reduces maintenance costs.

The global NAT policy contains NAT rules. A NAT rule contains the following elements:

- **Packet match criteria**—The packet match criteria can match packets by source IP address, destination IP address, service type, source security zone, or destination security zone. You can configure different packet match criteria for different NAT rules. The device translates the IP addresses of the matching packets. A matching packet refers to a packet that matches all match criteria in a NAT rule.
- **Action**—Action to take on matching packets, which can be source address translation (SNAT) or destination address translation (DNAT). SNAT can hide the IP addresses of internal hosts to external devices. DNAT is commonly used for internal servers to provide services for external users. A combination of SNAT and DNAT translates the source and destination IP addresses of packets.

NAT rules include the following types:

- **SNAT rule**—Used for source address translation.
- **DNAT rule**—Used for destination address translation.
- **SNAT+ DNAT rule**—Used for both source address translation and destination address translation.

Global NAT tasks at a glance

1. Configuring the global NAT policy
   a. Creating the global NAT policy
   b. Configuring NAT-type rules
   c. Configuring NAT64-type rules
   d. Configuring NAT66-type rules
   e. Rearranging NAT rules in the policy to adjust their priority
   f. Disabling NAT rules
2. (Optional.) Configuring NAT ALG
3. (Optional.) Configuring NAT DNS mapping
4. (Optional.) Configuring high availability for NAT
   - Enabling NAT port halving
   - Configuring hot backup for NAT
5. (Optional.) Configuring NAT in specific networks
   - Enabling NAT reply redirection
   - Enabling the deletion of timestamps in TCP SYN and SYN ACK packets
6. (Optional.) Configuring NAT maintenance
   - Configuring periodic NAT statistics collection
   - Enabling statistics collection for NAT session creation rate
   - Specifying a probe method for detecting reachability of NAT address group members

   - Enabling sending ICMP error messages for NAT failures
   - Enabling NAT configuration changes to take effect only on new connections
7. (Optional.) Configuring NAT logging
   - Configuring NAT session logging
   - Configuring NAT444 user logging
   - Configuring NAT alarm logging
   - Enabling logging for IP usage of a NAT address group in NO-PAT mode

Configuring the global NAT policy

About the global NAT policy

The global NAT policy contains a set of NAT rules to identify and translate matching packets. Compared with interface-based NAT policies, you do not need to apply the global NAT policy to any interface.

The global NAT policy contains NAT rules. A NAT rule contains the following elements:

- **Packet match criteria**—The packet match criteria can match packets by source IP address, destination IP address, service type, source security zone, or destination security zone. You can configure different packet match criteria for different NAT rules. The device translates the IP addresses of the matching packets. A matching packet refers to a packet that matches all match criteria in a NAT rule.
- **Action**—Action to take on matching packets, which can be source address translation or destination address translation.

NAT rules include the following types:

- **NAT type**—Used for translation between IPv4 addresses. For more information about NAT, see *NAT Configuration Guide*.
- **NAT64 type**—Used for translation between IPv4 addresses and IPv6 addresses. For more information about NAT64, see *NAT Configuration Guide*.
- **NAT66 type**—Used for translation between IPv6 addresses or translation between IPv6 address prefixes. For more information about NAT66, see *NAT Configuration Guide*.

Restrictions and guidelines for global NAT policy configuration

If no object group or security zone is specified for a NAT rule, this rule matches all packets.

NAT rules in the global NAT policy take effect only when the **Config status** of the policy is **Active**. You can use the **`display nat global-policy`** command to verify the status of the global policy.

- When the status is **Active**, the NAT rules in the global NAT policy are sorted in their configuration order. A rule configured earlier has a higher priority. The matching process stops when a packet matches a NAT rule. You can use the **`display this`** command to view the configuration order of the NAT rules.
- If the status is **Inactive**, the NAT rules are not used to match packets.

A maximum of 10000 NAT rules can be created for the global NAT policy.

Creating the global NAT policy

1. Enter system view

system-view

2. Create the global NAT policy and enter its view.

   nat global-policy

Configuring NAT-type rules

Restrictions and guidelines

When you configure a DNAT or SNAT+DNAT rule, follow these restrictions and guidelines:

- A destination security zone cannot be used as a match criterion.
- Only one destination address after NAT is supported:
  - You cannot execute the **action dnat ip-address** *local-address* [**local-port** *local-port*] command multiple times to configure multiple destination addresses after NAT.
  - Specify an object group that has only one IP address when you execute the **action dnat object-group** *ipv4-object-group-name* [**local-port** *local-port*] command.

For an object group-based static mapping to take effect, the object group cannot have excluded addresses.

Procedure

1. Enter system view.

   system-view

2. Enter the global NAT policy view.

   nat global-policy

3. Create a NAT-type rule and enter its view.

   rule name *rule-name* [**type nat**]

   By default, no NAT rule exists.

4. (Optional.) Configure a description for the NAT rule.

   description *text*

   By default, no description is configured for the NAT rule.

5. Specify packet match criteria for the rule.
   - Specify a source IP address match criterion.

     source-ip { *object-group-name* | **host** *ip-address* | **subnet** *subnet-ip-address mask-length* }

     By default, no source IP address match criterion is specified for the NAT rule.
   - Specify a destination IP address match criterion.

     destination-ip { *object-group-name* | **host** *ip-address* | **subnet** *subnet-ip-address mask-length* }

     By default, no destination IP address match criterion is specified for the NAT rule.
   - Specify a service object group.

     service *object-group-name*

     By default, no service type is specified for the NAT rule.
   - Specify a source security zone.

     source-zone *source-zone-name*

     By default, no source security zone is specified for the NAT rule.
   - Specify a destination security zone.

destination-zone *destination-zone-name*

By default, no destination security zone is specified for the NAT rule.

- Specify a VPN instance.

  vrf *vrf-name*

  By default, no VPN instance is specified for the NAT rule.

6. Specify an address translation method for the NAT rule.
   - Specify a source address translation method.

     NO-PAT:

     action snat { **address-group** { *group-id* | **name** *group-name* } | **object-group** *ipv4-object-group-name* } **no-pat** [**reversible**] [**vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     PAT:

     action snat { **address-group** { *group-id* | **name** *group-name* } | **object-group** *ipv4-object-group-name* } [**port-preserved**] [**vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     Easy IP:

     action snat easy-ip [**port-preserved**] [**vrf** *vrf-name*]

     Static translation:

     action snat static { **ip-address** *global-address* | **object-group** *object-group-name* | **subnet** *subnet-ip-address mask-length* } [**vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     NO-NAT:

     action snat no-nat

     By default, no source address translation method is specified for the NAT rule.
   - Specify a destination address translation method.

     Server mapping:

     action dnat { **ip-address** *local-address* | **object-group** *ipv4-object-group-name* } [**local-port** *local-port*] [**vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     NO-NAT:

     action dnat no-nat

     By default, no destination address translation method is specified for the NAT rule.
7. (Optional.) Enable hit counting for the NAT rule.

   counting enable

   By default, hit counting is disabled for the NAT rule.
8. Specify a translation mode for PAT.

   a. Return to global NAT policy view

   quit

   b. Return to system view.

   quit

   c. Apply the Endpoint-Independent Mapping mode for address translation.

   nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   By default, the Address and Port-Dependent Mapping mode applies.

   This command takes effect only on outbound PAT.

Configuring NAT64-type rules

About this task

NAT64-type rules use the AFT function, and are applicable to the following scenarios:

- **Access initiated on the IPv6 side**—In the earlier stage of transition from IPv4 to IPv6, most services are in IPv4 networks. When an IPv6 network user accesses an IPv4 network service, both the source address and destination address of user packets are IPv6 addresses. In this case, the source address and destination address must be translated to IPv4 addresses.
- **Access initiated on the IPv4 side**—In the later stage of transition from IPv4 to IPv6, most services are in IPv6 networks. When an IPv4 network user accesses an IPv6 network service, both the source address and destination address of user packets are IPv4 addresses. In this case, the source address and destination address must be translated to IPv6 addresses.

In the scenarios above, both source address translation and destination address translation are required.

Restrictions and guidelines

If you configure multiple packet match criteria in a NAT64-type rule, the type of IP addresses in the later configured packet match criteria must be the same as that in the earlier configured packet match criteria. For example, if you first execute the **`source-ip host 192.168.1.1`** command, the **`source-ip host 100::1`** command later executed does not take effect. Select an IP type as needed.

When you use the prefix method for address translation, the IPv6 address prefix length in the match criteria must meet the general prefix, IVI prefix, or NAT64 prefix requirements in the NAT action.

Procedure

1. Enter system view.

   `system-view`

2. Enter global NAT policy view.

   `nat global-policy`

3. Create a NAT64-type rule and enter its view.

   `rule name` *`rule-name`* **`type nat64`**

   By default, no NAT rule exists.

4. (Optional.) Configure a description for the NAT rule.

   `description` *`text`*

   By default, no description is configured for the NAT rule.

5. Specify packet match criteria for the rule.

   - Specify a source IP address match criterion.

     `source-ip` { { *ipv4-object-group-name* | *ipv6-object-group-name* } | **host** { *ipv4-address* | *ipv6-address* } | **subnet** { *subnet-ipv4-address mask-length* | *subnet-ipv6-address prefix-length* } }

     By default, no source IP address match criterion is specified for the NAT rule.

   - Specify a destination IP address match criterion.

     `destination-ip` { { *ipv4-object-group-name* | *ipv6-object-group-name* } | **host** { *ipv4-address* | *ipv6-address* } | **subnet** { *subnet-ipv4-address mask-length* | *subnet-ipv6-address prefix-length* } }

   - Specify a service object group.

     `service` *`object-group-name`*

By default, no service object group is specified for the NAT rule.

- Specify a source security zone.

  source-zone *source-zone-name*

  By default, no source security zone is specified for the NAT rule.

- Specify a VPN instance.

  vrf *vrf-name*

  By default, no VPN instance is specified for the NAT rule.

6. Specify an address translation method for the NAT rule.

   - Specify a source address translation method.

     NO-PAT:

     action snat object-group *ipv4-object-group-name* **no-pat** [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     action snat object-group *ipv6-object-group-name* **no-pat** [**vrf** *vrf-name*]

     PAT:

     action snat object-group *ipv4-object-group-name* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     action snat object-group *ipv6-object-group-name* [**vrf** *vrf-name*]

     Prefix translation:

     action snat prefix { **general** { **v4tov6** *prefix-general general-prefix-length* | **v6tov4** } | **ivi v6tov4** | **nat64 v4tov6** *prefix-nat64 nat64-prefix-length* } [**vrf** *vrf-name*]

     Static translation:

     action snat static ip-address *global-ipv4-addres* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

   - Specify a destination address translation method.

     Static translation:

     action dnat static ip-address *local-ipv4-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     action dnat static ip-address *local-ipv6-address* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     Server mapping:

     action dnat server ip-address *local-ipv4-address* [**port** *local-port*] [**vrf** *vrf-name*]

     action dnat server ip-address *local-ipv6-address* [**local-port** *local-port*] [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

     Prefix method:

     action dnat prefix { **general v6tov4** | **nat64 v6tov4** } [**vrf** *vrf-name*]

     action dnat prefix { **general v4tov6** *prefix-general prefix-length* | **ivi v4tov6** *prefix-ivi* } [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

7. (Optional.) Enable hit counting for the NAT rule.

   counting enable

   By default, hit counting is disabled for the NAT rule.

Configuring NAT66-type rules

About this task

NAT66 is used for translation between IPv6 addresses. Available translation methods include dynamic, static, and NPTv6. An IPv6 address includes a network prefix and an interface identifier. The NPTv6 method translates the network prefix to a new network prefix, and the interface identifier of the IPv6 address will be adjusted according to RFC 6296. For more information about the algorithm, see RFC 6296. The other two methods translate an IPv6 address to a new IPv6 address. When a large number of IPv6 addresses exist and you are not sensitive to the addresses after translation, as a best practice, use the NPTv6 method.

Restrictions and guidelines

For a DNAT or SNAT+DNAT rule, a destination security zone cannot be used as a match criterion..

Procedure

1. Enter system view.

   `system-view`

2. Enter global NAT policy view.

   `nat global-policy`

3. Create a NAT66-type rule and enter its view.

   `rule name` *`rule-name`* **`type nat66`**

   By default, no NAT rule exists.

4. (Optional.) Configure a description for the NAT rule.

   `description` *`text`*

   By default, no description is configured for the NAT rule.

5. Specify packet match criteria for the rule.

   - Specify a source IP address match criterion.

     `source-ip` `{` *`ipv6-object-group-name`* `|` **`host`** *`ipv6-address`* `|` **`subnet`** *`subnet-ipv6-address prefix-length`* `}`

     By default, no source IP address match criterion is specified for the NAT rule.

   - Specify a destination IP address match criterion.

     `destination-ip` `{` *`ipv6-object-group-name`* `|` **`host`** *`ipv6-address`* `|` **`subnet`** *`subnet-ipv6-address prefix-length`* `}`

   - Specify a service object group.

     `service` *`object-group-name`*

     By default, no service object group is specified for the NAT rule.

   - Specify a source security zone.

     `source-zone` *`source-zone-name`*

     By default, no source security zone is specified for the NAT rule.

   - Specify a destination security zone.

     `destination-zone` *`destination-zone-name`*

     By default, no destination security zone is specified for the NAT rule.

   - Specify a VPN instance.

     `vrf` *`vrf-name`*

     By default, no VPN instance is specified for the NAT rule.

6. Specify an address translation method for the NAT rule.

   - Specify a source address translation method.

NO-PAT:

action snat object-group *ipv6-object-group-name* **no-pat** [**vrf** *vrf-name*]

PAT:

action snat object-group *ipv6-object-group-name* [**vrf** *vrf-name*]

Static translation:

action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

NPTv6:

action snat nptv6 *translated-ipv6-prefix nptv6-prefix-length* [**vrf** *vrf-name*]

NO-NAT:

action snat no-nat

- Specify a destination address translation method.

  Server mapping:

  action dnat ip-address *local-ipv6-address* [**local-port** *local-port*] [**vrf** *vrf-name*]

  NPTv6:

  action dnat nptv6 *translated-ipv6-prefix nptv6-prefix-length* [**vrf** *vrf-name*]

  NO-NAT:

  action dnat no-nat

7. (Optional.) Enable hit counting for the NAT rule.

   counting enable

   By default, hit counting is disabled for the NAT rule.

Rearranging NAT rules in the policy to adjust their priority

About this task

In the global NAT policy, the priority of NAT rules are determined by the configuration order. A rule configured earlier has a higher priority. You can use the **rule move** command to rearrange the NAT rules to adjust their priority.

Restrictions and guidelines

You can use this feature to rearrange only existing NAT rules to change their priority.

When you rearrange global NAT rules to change their priority, make sure all NAT rules containing destination address translation methods are before the NAT rules containing only source address translation methods.

- Do not place a NAT rule containing a destination address translation method after a NAT rule containing only a source address translation method.
- Do not place a NAT rule containing only a source address translation method before a NAT rule containing a destination address translation method.

Procedure

1. Enter system view.

   system-view

2. Enter the global NAT policy view.

nat global-policy

3. Rearrange NAT rules to change their priority.

   rule move *rule-name1* [**type** { **nat** | **nat64** | **nat66** }] { **after** | **before** } [*rule-name2* [**type** { **nat** | **nat64** | **nat66** }]]

Disabling NAT rules

Restrictions and guidelines

This feature does not delete a NAT rule, but makes the rule ineffective. To delete a NAT rule, use the **undo rule name** command.

Procedure

1. Enter system view.

   system-view

2. Enter the global NAT policy view.

   nat global-policy

3. Enter the view of a NAT rule.

   rule name *rule-name* [**type** { **nat** | **nat64** | **nat66** }]

4. Disable the NAT rule.

   disable

   By default, NAT rules are enabled.

Configuring NAT ALG

1. Enter system view.

   system-view

2. Configure NAT ALG for a protocol or all protocols.

   nat alg { **all** | **dns** | **ftp** | **h323** | **icmp-error** | **ils** | **mgcp** | **nbt** | **pptp** | **rsh** | **rtsp** | **sccp** | **sctp** | **sqlnet** | **tftp** | **xdmcp** }

   By default, NAT ALG is enabled for DNS, FTP, ICMP error messages, PPTP, and RTSP, and is disabled for the other supported protocols.

Configuring NAT DNS mapping

About this task

NAT DNS mapping is applicable to scenarios that internal users access an internal server by using the domain name of the server when the DNS server is located at the external network.

Enabled with this feature, NAT translates the public IP address in the DNS reply payload into the private IP address. So that the internal users access the internal server by using the private IP address.

NAT DNS mapping works in conjunction with address translation of the server mapping method:

1. A NAT DNS mapping maps the domain name to the public IP address, public port number, and the protocol type for the internal server.
2. To specify the mapped public IP address, configure the destination IP address match criterion. To specify the mapped protocol type and public port number, configure the service type match criterion. To specify the translated private IP address, perform the address translation.

3. Upon receiving the DNS response, the NAT device performs a DNS mapping lookup by using the domain name in the response. If a match is found, NAT translates the public IP address in the response into the private IP address.

Procedure

1. Enter system view.

   system-view

2. Enable NAT ALG for DNS.

   nat alg dns

   By default, NAT ALG is enabled for DNS.

3. Configure a NAT DNS mapping.

   nat dns-map domain *domain-name* **protocol** *pro-type* { **interface** *interface-type interface-number* | **ip** *global-ip* } **port** *global-port*

   You can configure multiple NAT DNS mappings.

Enabling NAT port halving

About this task

After you enable NAT port halving in VRRP load balancing on an IRF fabric, each port block will be equally divided between the two devices. The two devices will use different ports to translate packets from the same IP address, avoiding port assignment conflicts.

For more information about the IRF fabric, see *Virtual Technologies Configuration Guide*.

Restrictions and guidelines

Do not use this feature in VRRP standard mode on an IRF fabric.

Procedure

1. Enter system view.

   system-view

2. Enable NAT port halving.

   nat port-load-balance enable slot *slot-number*

   By default, NAT port halving is disabled.

Configuring hot backup for NAT

About hot backup for NAT

If only one NAT device is deployed in the internal network, internal users cannot access the external network when the NAT device fails. To avoid this situation, configure hot backup for NAT. In the hot backup plan, the two devices in a hot backup system are capable of processing NAT services. Session entries, session relation entries, NAT port block entries, and NAT configurations are synchronized through the hot backup channel. When one device fails, the other device takes over.

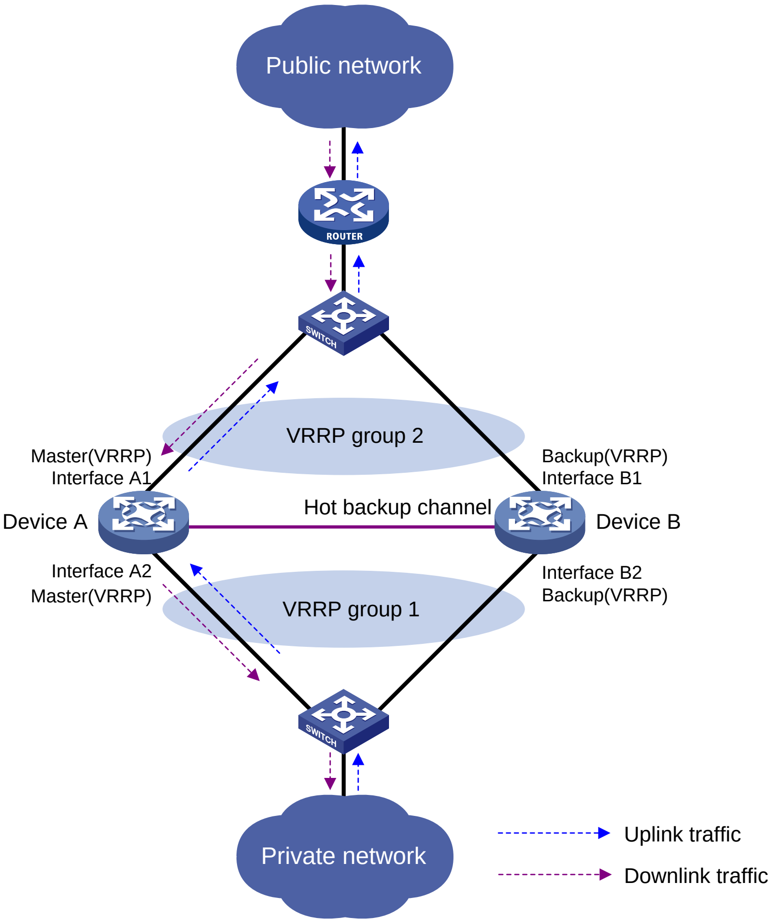
For more information about configuring hot backup, see *High Availability Configuration Guide*.

Operating mechanism

Typically, the master device in the VRRP group processes NAT services in the hot backup system. The following example illustrates how the hot backup system in active/standby mode ensures uninterrupted NAT services when the master device fails.

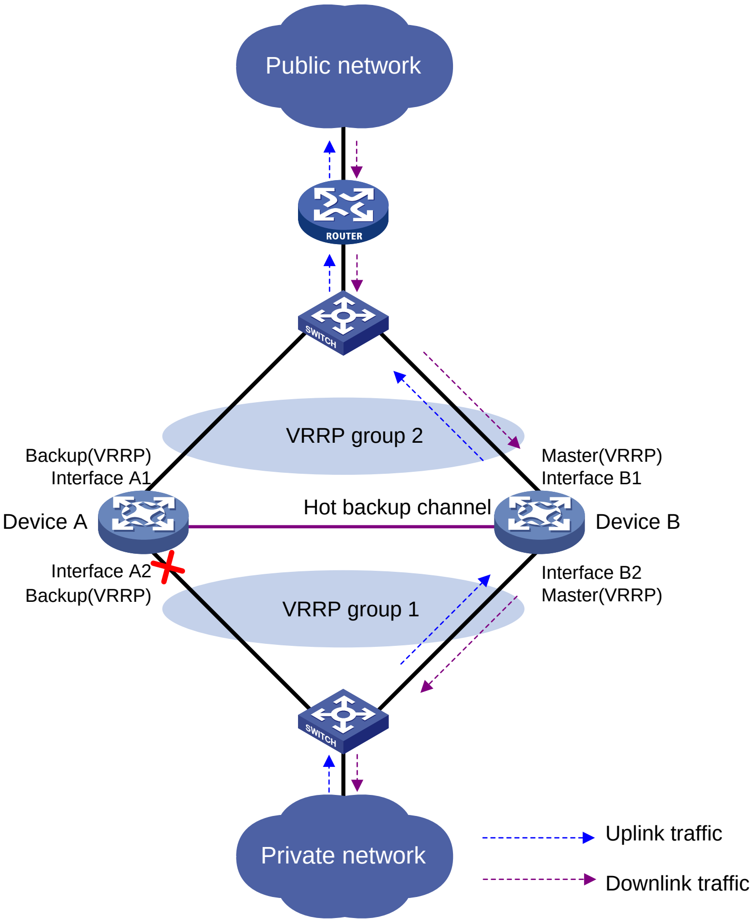
As shown in Figure 9, Device A acts as the primary device and Device B acts as the secondary device in a hot backup system. Device A synchronizes its session entries, session relation entries, and port block entries to Device B in real time through the hot backup channel. Downlinks of Device A and Device B are in VRRP group 1 and uplinks of Device A and Device B are in VRRP group 2. VRRP groups are associated with the hot backup system. RBM selects Device A as the master device for address translation based on the link status or forwarding capability of Device A.

Figure 9 Hot backup system in active/standby mode



As shown in Figure 10, when Interface A2 of Device A fails, Device B becomes the master device in the VRRP group. Because Device B has NAT configuration information and service entries, NAT services are not interrupted after link switchover.

Figure 10 Traffic switchover in active/standby mode



Configuring the hot backup system in active/standby mode

About this task

For active/standby hot backup, some translation rules for static address translation, source address translation, and destination address translation assign the public address after translation or public IP address of the internal server to the address management module. Then, both the active and standby devices advertise the mappings between the public IP address and MAC addresses of their own physical interfaces to all nodes in the same LAN or local link. As a result, the uplink Layer 3 device directly connected to the hot backup system might send downlink packets to the backup device, causing service anomalies.

To avoid such an issue, bind an address translation method to a VRRP group. Then, only the master device responds to ARP requests or NS messages with the virtual MAC address of the VRRP group. For more information about configuring the hot backup system, see *High Availability Configuration Guide*.

Restrictions and guidelines

Bind an address translation method to a VRRP group in NAT rule view on the primary device in the hot backup system. The virtual IP address of the VRRP group and the public IP address after translation or the public IP address of the internal server are on the same segment.

Procedure (NAT-type rule)

1. Enter system view.

system-view

2. Enter global NAT policy view.

   nat global-policy

3. Enter NAT rule view.

   rule name *rule-name* [**type nat**]

4. Bind a translation method to a VRRP group. Choose the options to configure as needed:
   - Bind the NO-PAT method to a VRRP group for source address translation.

     action snat address-group { *group-id* | **name** *group-name* } **no-pat** [**reversible**] **vrrp** *virtual-router-id*
   - Bind the PAT method to a VRRP group for source address translation.

     action snat address-group { *group-id* | **name** *group-name* } [**port-preserved**] **vrrp** *virtual-router-id*
   - Bind the static source address translation method to a VRRP group.

     action snat static { **ip-address** *global-address* | **object-group** *object-group-name* | **subnet** *subnet-ip-address mask-length* } **vrrp** *virtual-router-id*
   - Bind the server mapping method to a VRRP group for destination address translation.

     action dnat { **ip-address** *local-address* | **object-group** *ipv4-object-group-name* } [**local-port** *local-port*] **vrrp** *virtual-router-id*

   By default, a translation method is not bound to any VRRP group.

Procedure (NAT64-type rule)

1. Enter system view.

   system-view

2. Enter global NAT policy view.

   nat global-policy

3. Create a NAT64-type rule and enter its view.

   rule name *rule-name* **type nat64**

4. Bind a translation method to a VRRP group. Choose the options to configure as needed:
   - Bind the NO-PAT method to a VRRP group for source address translation.

     action snat object-group *ipv4-object-group-name* **no-pat** [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the PAT method to a VRRP group for source address translation.

     action snat object-group *ipv4-object-group-name* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv6-to-IPv4 static source address translation method to a VRRP group.

     action snat static ip-address *global-ipv4-address* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv4-to-IPv6 static source address translation method to a VRRP group.

     action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv6-to-IPv4 static destination address translation method to a VRRP group.

     action dnat static ip-address *local-ipv4-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv4-to-IPv6 static destination address translation method to a VRRP group.

action dnat static ip-address *local-ipv6-address* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

- Bind the server mapping method to a VRRP group for IPv4-to-IPv6 destination address translation.

  action dnat server ip-address *local-ipv6-address* [**local-port** *local-port*] [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
- Bind the prefix method to a VRRP group for IPv4-to-IPv6 destination address translation.

  action dnat prefix { **general v4tov6** *prefix-general prefix-length* | **ivi v4tov6** *prefix-ivi* } [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

By default, a translation method is not bound to any VRRP group.

Procedure (NAT66-type rule)

1. Enter system view.

   system-view
2. Enter global NAT policy view.

   nat global-policy
3. Create a NAT66-type rule and enter its view.

   rule name *rule-name* **type nat66**
4. Bind the static source address translation method to a VRRP group.

   action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

   By default, a static source address translation method is not bound to any VRRP group.

Configuring the hot backup system in dual-active mode

About this task

For dual-active hot backup, the uplink Layer 3 device directly connected to the hot backup system might send downlink packets to the backup device, causing service anomalies.

To avoid such an issue, bind an address translation method to a VRRP group. Then, only the master device responds to ARP requests or NS messages with the virtual MAC address of the VRRP group. For more information about configuring the hot backup system, see *High Availability Configuration Guide*.

Restrictions and guidelines

Select one of the following configuration methods:

- The two devices can share the same NAT address group. To prevent different master devices from using the same IP-port mapping for different hosts, specify the PAT translation mode for NAT rules and execute the **nat remote-backup port-alloc** command on the primary device.
- As a best practice to prevent different master devices from using the same IP-port mapping for different hosts, configure the two devices to use different public IP addresses for address translation. For example, if the two devices use addresses in different NAT address groups, user traffic with different source IP addresses is identified by source IP address match criteria in NAT rules. To enable different master devices to translate the forward user traffic, specify different gateway addresses for different internal users. To direct the reverse traffic to different master devices, bind the address translation method to different VRRP groups on the primary device for load sharing.

Bind an address translation method to a VRRP group in NAT rule view on the primary device in the hot backup system. The virtual IP address of the VRRP group and the public IP address after translation or public IP address of the internal server are on the same segment.

Procedure (NAT-type rule)

1. Enter system view.
   ```
   system-view
   ```
2. Enter global NAT policy view.
   ```
   nat global-policy
   ```
3. Enter NAT rule view.
   ```
   rule name rule-name [ type nat ]
   ```
4. Bind a translation method to a VRRP group. Choose the options to configure as needed:
   - Bind the NO-PAT method to a VRRP group for source address translation.
     ```
     action snat address-group { group-id | name group-name } no-pat
     [ reversible ] vrrp virtual-router-id
     ```
   - Bind the PAT method to a VRRP group for source address translation.
     ```
     action snat address-group { group-id | name group-name }
     [ port-preserved ] vrrp virtual-router-id
     ```
   - Bind the static source address translation method to a VRRP group.
     ```
     action snat static { ip-address global-address | object-group
     object-group-name | subnet subnet-ip-address mask-length } vrrp
     virtual-router-id
     ```
   - Bind the server mapping method to a VRRP group for destination address translation.
     ```
     action dnat { ip-address local-address | object-group
     ipv4-object-group-name } [ local-port local-port ] vrrp
     virtual-router-id
     ```

   By default, no translation method is bound to any VRRP group.
5. (Optional.) Specify NAT port block ranges for the two devices in the hot backup system.
   a. Return to global NAT policy view.
      ```
      quit
      ```
   b. Return to system view.
      ```
      quit
      ```
   c. Specify NAT port ranges for the two devices in the hot backup system.
      ```
      nat remote-backup port-alloc { primary | secondary }
      ```
      By default, the two devices in the hot backup system share NAT port resources.

      The following table describes port ranges indicated by the keywords:

Keyword	Port ranges
primary	The first half of the port range.
secondary	The second half of the port range.

Procedure (NAT64-type rule)

1. Enter system view.
   ```
   system-view
   ```
2. Enter global NAT policy view.
   ```
   nat global-policy
   ```

3. Create a NAT64-type rule and enter its view.

   rule name *rule-name* **type nat64**

4. Bind a translation method to a VRRP group. Choose the options to configure as needed:
   - Bind the NO-PAT method to a VRRP group for source address translation.

     action snat object-group *ipv4-object-group-name* **no-pat** [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the PAT method to a VRRP group for source address translation.

     action snat object-group *ipv4-object-group-name* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv6-to-IPv4 static source address translation method to a VRRP group.

     action snat static ip-address *global-ipv4-address* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv4-to-IPv6 static source address translation method to a VRRP group.

     action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv6-to-IPv4 static destination address translation method to a VRRP group.

     action dnat static ip-address *local-ipv4-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the IPv4-to-IPv6 static destination address translation method to a VRRP group.

     action dnat static ip-address *local-ipv6-address* [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the server mapping method to a VRRP group for IPv4-to-IPv6 destination address translation.

     action dnat server ip-address *local-ipv6-address* [**local-port** *local-port*] [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]
   - Bind the prefix method to a VRRP group for IPv4-to-IPv6 destination address translation.

     action dnat prefix { **general v4tov6** *prefix-general prefix-length* | **ivi v4tov6** *prefix-ivi* } [**ipv4-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

   By default, a translation method is not bound to any VRRP group.

Procedure (NAT66-type rule)

1. Enter system view.

   system-view

2. Enter global NAT policy view.

   nat global-policy

3. Create a NAT66-type rule and enter its view.

   rule name *rule-name* **type nat66**

4. Bind the static source address translation method to a VRRP group.

   action snat static ip-address *global-ipv6-address* [**ipv6-vrrp** *virtual-router-id*] [**vrf** *vrf-name*]

   By default, a static source address translation method is not bound to any VRRP group.

Configuring NAT in specific networks

Enabling NAT reply redirection

About this task

In some network scenarios, the inbound dynamic NAT is configured with tunneling, and multiple tunnel interfaces use the same NAT address group. In this case, the device will translate the source IP addresses of packets from different tunnels into the same NAT address before forwarding them. When the forwarding interface receives the reply packets, the device, by default, will not look up the NAT session table. This will cause the incorrect forwarding of the reply packets. To solve the problem, you can enable the NAT reply redirection feature on the forwarding interface. NAT reply redirection allows the interface to use the NAT session table to translate the destination IP addresses for NAT reply packets and find the correct output interfaces for those NATed reply packets.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable NAT reply redirection.

   `nat redirect reply-route enable`

   By default, NAT reply redirection is disabled.

Enabling the deletion of timestamps in TCP SYN and SYN ACK packets

About this task

With this feature configured, the system deletes the timestamps from the TCP SYN and SYN ACK packets after dynamic address translation.

If PAT mode is configured on an interface by using **`nat inbound`** or **`nat outbound`**, and the tcp_timestams and tcp_tw_recycle function is configured on the TCP server, TCP connections might not be established. To solve the problem, you can shut down the tcp_tw_recycle function or configure the **`nat timestamp delete`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enable the deletion of timestamps in TCP SYN and SYN ACK packets

   `nat timestamp delete` `[` **`vpn-instance`** *`vpn-instance-name`* `]`

   By default, the deletion of timestamps in TCP SYN and SYN ACK packets is disabled.

   You can enable this feature for multiple VPN instances by repeating the command with different VPN parameters.

Configuring NAT maintenance

Configuring periodic NAT statistics collection

About this task

This feature periodically counts sessions and port block assignment failures for address groups.

Hardware and feature compatibility

Models	Feature compatibility
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280	No
NFNX5-HD6480, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	Yes

Restrictions and guidelines

This feature might cause intensive CPU usage. You can disable the feature when CPU resources are insufficient.

Procedure

1. Enter system view.

   `system-view`

2. Enable periodic NAT statistics collection.

   `nat periodic-statistics enable`

   By default, periodic NAT statistics collection is disabled.

3. Configure the interval for periodic NAT statistics collection.

   `nat periodic-statistics interval` *`interval`*

   By default, the interval for periodic NAT statistics collection is 300 seconds.

   A narrower interval indicates intensive CPU usage. As a best practice, use the default interval value.

Enabling statistics collection for NAT session creation rate

About this task

This feature collects information about NAT session creation rates. To view the statistics, use the **`display nat statistics`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enable statistics collection for NAT session creation rate.

   `nat session create-rate enable`

   By default, statistics collection for NAT session creation rate is disabled.

Specifying a probe method for detecting reachability of NAT address group members

About this task

The NAT address group probing uses an NQA template to detect the reachability of the addresses in the group. For information about NQA, see *Network Management and Monitoring Configuration Guide*.

The device periodically sends probe packets to the specified destination address in the NQA template. The source IP addresses in the probe packets are the IP addresses in the NAT address group.

- If the device receives a response packet for a probe, the probed source IP address can be used for address translation.
- If the device does not receive a response packet for a probe, the probed source IP address will be excluded from address translation temporarily. However, in the next NQA operation period, this excluded IP address is also probed. If a response is received in this round, the IP address can be used for address translation.

Restrictions and guidelines

You can specify multiple NQA templates in one NAT address group view. An IP address in the address group is identified as reachable as long as one probe for this IP address succeeds.

This feature is applicable to NAT address groups used for outbound address translation. The manually configured excluded IP addresses are not probed.

Make sure the NQA template used for NAT address group probing does not have source IP address configured.

Procedure

1. Enter system view.

   system-view

2. Enter NAT address group view.

   nat address-group *group-id* [**name** *group-name*]

3. Specify a probe method for the NAT address group.

   probe *template-name*

   By default, no probe method is specified for a NAT address group.

   You can specify a nonexistent probe method. The probing takes effect only after you create and configure the NQA template.

Enabling sending ICMP error messages for NAT failures

About this task

By default, the device does not send ICMP error messages when NAT fails. Disabling sending ICMP error messages for NAT failures reduces useless packets, saves bandwidth, and avoids exposing the firewall IP address to the public network.

Procedure

1. Enter system view.

   system-view

2. Enable sending ICMP error messages for NAT failures.

   nat icmp-error reply

By default, no ICMP error messages are sent for ICMP packet translation failures.

Enabling NAT configuration changes to take effect only on new connections

About this task

By default, NAT configuration changes (such as adding, deleting, editing, or moving NAT rules) might cause traffic on an established connection to match a new NAT rule. As a result, you must create a new connection.

Execute the **`nat configuration-for-new-connection enable`** command if you do not want the NAT configuration change to affect existing connections. After you execute this command on the device, it still performs address translation according to the NAT rules before the configuration change for traffic on existing connections. For traffic on new connections, the device matches the traffic according to the priority of NAT rules after the configuration change and performs address translation based on the matching NAT rules.

Procedure

1. Enter system view.

   `system-view`

2. Enable NAT configuration change to take effect only on new connections.

   `nat configuration-for-new-connection enable`

   By default, NAT configuration change taking effect only on new connections is disabled.

Configuring NAT logging

Configuring NAT session logging

About this task

NAT session logging records NAT session information, including translation information and access information.

A NAT device generates NAT session logs for the following events:

- NAT session establishment.
- NAT session removal. This event occurs when you add a configuration with a higher priority, remove a configuration, change ACLs, when a NAT session ages out, or when you manually delete a NAT session.
- Active NAT session logging.

Procedure

1. Enter system view.

   `system-view`

2. Enable NAT logging.

   `nat log enable` [**`acl`** { *`ipv4-acl-number`* | **`name`** *`ipv4-acl-name`* }]

   By default, NAT logging is disabled.

3. Enable NAT session logging.
   - For NAT session establishment events:

     `nat log flow-begin`

   - For NAT session removal events:

```
nat log flow-end
```

- For active NAT flows:

```
nat log flow-active time-value
```

By default, NAT session logging is disabled.

Configuring NAT444 user logging

About this task

NAT444 user logs are used for user tracing. The NAT444 gateway generates a user log whenever it assigns or withdraws a port block. The log includes the private IP address, public IP address, and port block. You can use the public IP address and port numbers to locate the user's private IP address from the user logs.

A NAT444 gateway generates NAT user logs when one of the following events occurs:

- A port block is assigned.

  For the NAT444 static port block mapping, the NAT444 gateway generates a user log when it translates the first connection from a private IP address.

  For the NAT444 dynamic port block mapping, the NAT444 gateway generates a user log when it assigns or extends a port block for a private IP address.

- A port block is withdrawn.

  For the NAT444 static port block mapping, the NAT444 gateway generates a user log when all connections from a private IP address are disconnected.

  For the NAT444 dynamic port block mapping, the NAT444 gateway generates a user log when all the following conditions are met:

  - All connections from a private IP address are disconnected.
  - The port blocks (including the extended ones) assigned to the private IP address are withdrawn.
  - The corresponding mapping entry is deleted.

Prerequisites

Before configuring NAT444 user logging, you must configure the custom NAT444 log generation and outputting features. For more information, see the information center in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enable NAT logging.

```
nat log enable [ acl { ipv4-acl-number | name ipv4-acl-name } ]
```

   By default, NAT logging is disabled.

   The **acl** keyword does not take effect on NAT444 user logging.

3. Enable NAT444 user logging. Choose the options to configure as needed:
   - For port block assignment:

```
nat log port-block-assign
```

   - For port block withdrawal:

```
nat log port-block-withdraw
```

   By default, NAT444 user logging is disabled.

Configuring NAT alarm logging

About this task

Packets that need to be translated are dropped if the NAT resources are not enough. In NO-PAT, the NAT resources refer to the public IP addresses. In EIM PAT, the NAT resources refer to public IP addresses and ports. In NAT444, the NAT resources refer to public IP addresses, port blocks, or ports in port blocks. NAT alarm logging monitors the usage of NAT resources and outputs logs if the NAT resources are not enough.

For NAT444 dynamic port block mappings, an alarm log is generated upon the port block assignment failure or the failure that port resources cannot meet the user address translation requirement.

Restrictions and guidelines

The **`nat log alarm`** command take effect only after you use the **`nat log enable`** command to enable NAT logging.

Prerequisites

Before configuring NAT alarm logging, you must configure the custom NAT log generation and outputting features. For more information, see the information center in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable NAT logging.

   `nat log enable` [**`acl`** { *`ipv4-acl-number`* | **`name`** *`ipv4-acl-name`* }]

   By default, NAT logging is disabled.

   The **`acl`** keyword does not take effect on NAT alarm logging.

3. Enable NAT alarm logging.

   `nat log alarm`

   By default, NAT alarm logging is disabled.

   An NAT alarm log is output when NAT resources run out.

4. (Optional.) Set the NAT444 port block usage threshold.

   `nat log port-block usage threshold` *`threshold-value`*

   By default, the NAT444 port block usage threshold is 90%.

   The system generates alarm logs if the port block usage exceeds the threshold.

Enabling logging for IP usage of a NAT address group in NO-PAT mode

About this task

The system generates a log if the IP usage of a NAT address group exceeds the threshold.

Restrictions and guidelines

This feature takes effect only after you enable NAT logging by using the **`nat log enable`** command.

Procedure

1. Enter system view.

system-view

2. Enable NAT logging.

   nat log enable [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   By default, NAT logging is disabled.

   The **acl** keyword does not take effect on the logging for IP usage of a NAT address group in NO-PAT mode.

3. Enable logging for the IP usage of a NAT address group in NO-PAT mode and set a threshold.

   nat log no-pat ip-usage [**threshold** *value*]

By default, logging is disabled for the IP usage of a NAT address group.

Display and maintenance commands for global NAT

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the NAT ALG status for all supported protocols.	**display nat alg**
Display all NAT configuration information.	**display nat all**
Display NAT address group information.	**display nat address-group** [*group-id*]
Display NAT logging configuration.	**display nat log**
Display information about NAT NO-PAT entries.	**display nat no-pat** [**slot** *slot-number*]
Display IP usage of NAT address groups in NO-PAT mode.	**display nat no-pat ip-usage** [**address-group** { *group-id* | **name** *group-name* } | **object-group** *object-group-name*] [**slot** *slot-number*]
Display periodic NAT statistics.	**display nat periodic-statistics** { **address-group** [*group-id* | **name** *group-name*] | **ip** *global-ip* } [**slot** *slot-number*]
Display NAT sessions.	**display nat session** [[**responder**] { **source-ip** *source-ip* | **destination-ip** *destination-ip* } * [**vpn-instance** *vpn -instance-name*]] [**slot**

	slot-number] [**brief** | **verbose**]
Display NAT statistics.	**display nat statistics** [**summary**] [**slot** *slot-number*]
Display NAT port block mappings.	**display nat port-block dynamic** [**address-group** { *group-id* | **name** *group-name* }] [**slot** *slot-number*]
Display the port block usage for address groups.	**display nat port-block-usage** [**address-group** *group-id*] [**slot** *slot-number*]
Display NAT address group probe information.	**display nat probe address-group** [*group-id*]
Clear NAT counting statistics.	**reset nat count statistics** { **all** | **dynamic** | **global-policy** | **server** | **static** }
Clear periodic NAT statistics.	**reset nat periodic-statistics** [**slot** *slot-number*]
Clear NAT sessions.	**reset nat session** [**slot** *slot-number*]

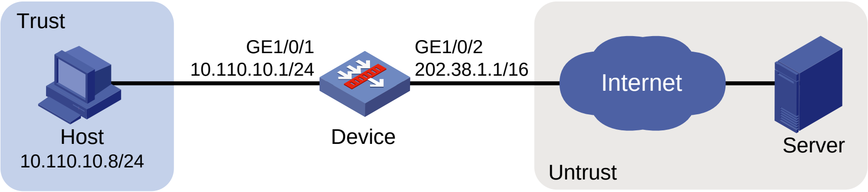
Global NAT configuration examples

Example: Configuring outbound one-to-one static NAT

Network configuration

Configure static NAT to allow the host at 10.110.10.8/24 to access the server at 201.20.1.1/24 on the Internet.

Figure 11 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   ［Device］ security-zone name trust
   ［Device-security-zone-Trust］ import interface gigabitethernet 1/0/1
   ［Device-security-zone-Trust］ quit
   ［Device］ security-zone name untrust
   ［Device-security-zone-Untrust］ import interface gigabitethernet 1/0/2
   ［Device-security-zone-Untrust］ quit
   ```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.2.

   ```
   [Device] ip route-static 201.20.1.0 24 202.38.1.2
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the host to the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-host 10.110.10.8
   [Device-security-policy-ip-1-trust-untrust] destination-ip-host 201.20.1.1
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure NAT:

   # Configure a one-to-one static NAT mapping between the private address 10.110.10.8 and the public address 202.38.1.100.

   ```
   [Device] nat global-policy
   [Device-nat-global-policy] rule name rule1
   [Device-nat-global-policy-rule-rule1] source-ip host 10.110.10.8
   [Device-nat-global-policy-rule-rule1] source-zone trust
   [Device-nat-global-policy-rule-rule1] destination-zone untrust
   [Device-nat-global-policy-rule-rule1] action snat static ip-address 202.38.1.100
   ```

Verifying the configuration

# Verify that the host at 10.110.10.8/24 can access the server on the Internet. (Details not shown.)

# Display static NAT configuration.

```
[Device] display nat global-policy
NAT global-policy information:
  Totally 1 NAT global-policy rules.
  Rule name: rule1
    Type                  : nat
    SrcIP address         : 10.110.10.8
```

```
    Source-zone name      : Trust
    Destination-zone name : Untrust
    SNAT action:
      Ipv4 address: 202.38.1.100
    NAT counting : 0
    Config status: Active
```

# Display NAT sessions.

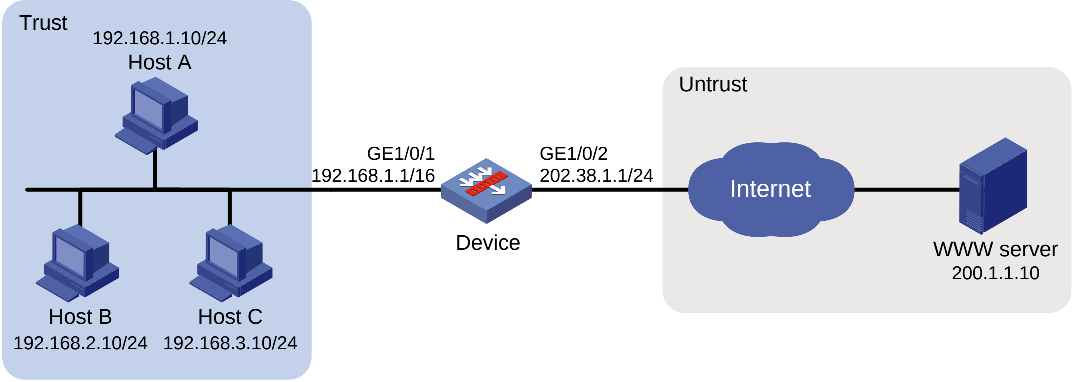
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.110.10.8/54765
  Destination IP/port: 201.20.1.1/23
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 201.20.1.1/23
  Destination IP/port: 202.38.1.100/54765
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: TELNET
Rule ID: -/-/-
Rule name:
Start time: 2017-05-19 10:57:47  TTL: 1195s
Initiator->Responder:           8 packets        375 bytes
Responder->Initiator:          10 packets        851 bytes

Total sessions found: 1
```

Example: Configuring outbound dynamic NAT (non-overlapping addresses)

Network configuration

As shown in Figure 12, a company has a private address 192.168.0.0/16 and two public IP addresses 202.38.1.2 and 202.38.1.3. Configure outbound dynamic NAT to allow only internal users on subnet 192.168.1.0/24 to access the Internet.

Figure 12 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 16
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.20.

```
[Device] ip route-static 200.1.1.0 24 202.38.1.20
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the hosts to the server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.1.1.10
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

   # Configure address group 0, and add an address range from 202.38.1.2 to 202.38.1.3 to the group.

```
[Device] nat address-group 0
[Device-address-group-0] address 202.38.1.2 202.38.1.3
```

```
[Device-address-group-0] quit
```

# Configure address object group **obj1** to identify packets from subnet 192.168.1.0/24.

```
[Device] object-group ip address obj1
[Device-obj-grp-ip-obj1] network subnet 192.168.1.0 24
[Device-obj-grp-ip-obj1] quit
```

# Configure a NAT rule for the global policy, and specify address object group **obj1** as the packet match criterion and use NAT address group **0** for source address and port translation.

```
[Device] nat global-policy
[Device-nat-global-policy] rule name rule1
[Device-nat-global-policy-rule-rule1] source-ip obj1
[Device-nat-global-policy-rule-rule1] action snat address-group 0
```

Verifying the configuration

# Verify that Host A can access the WWW server, while Host B or Host C cannot. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 0
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.3
    Exclude address information:
      Start address         End address
      ---                   ---

NAT global-policy information:
  Totally 1 NAT global-policy rules.
  Rule name: rule1
    Type                   : nat
    SrcIP object group     : obj1
  SNAT action:
      Address group ID: 0
      NO-PAT: N
      Reversible: N
      Port-preserved: N
    NAT counting : 0
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host A accesses the WWW server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.10/52082
  Destination IP/port: 200.1.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 200.1.1.10/80
  Destination IP/port: 202.38.1.2/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-19 16:16:59  TTL: 9995s
Initiator->Responder:         551 packets      32547 bytes
Responder->Initiator:         956 packets    1385514 bytes
Total sessions found: 1
```

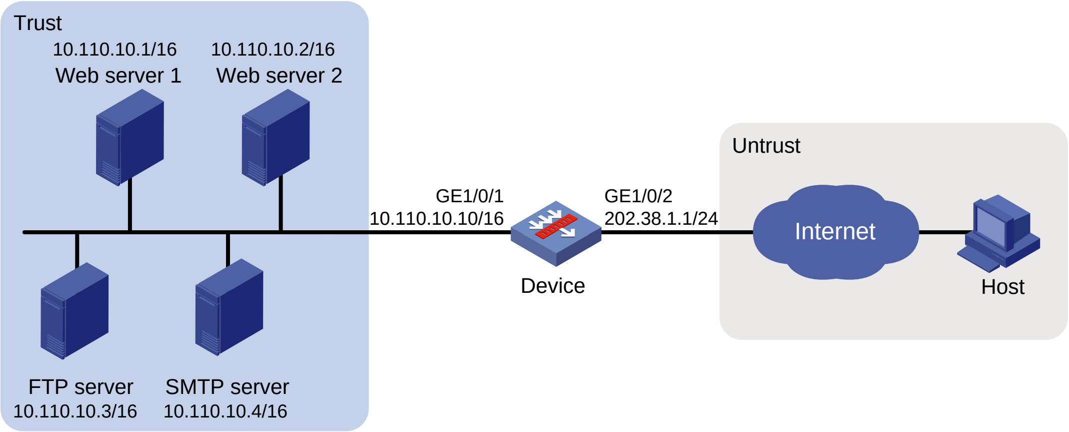
Example: Configuring NAT Server for external-to-internal access

Network configuration

As shown in Figure 13, two Web servers, one FTP server and one SMTP server, are in the internal network to provide services for external users. The internal network address is 10.110.0.0/16. The company has three public IP addresses from 202.38.1.1/24 to 202.38.1.3/24.

Configure the NAT Server feature to allow the external user to access the internal servers with public address 202.38.1.1/24.

Figure 13 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.10 16
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   ```

```
[Device-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **untrust-trust** to permit the packets from the host to the servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.1
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.2
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.3
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.4
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

4. Configure NAT:

# Configure service object groups for FTP, Web, and SMTP services.

```
[Device] object-group service service1
[Device-obj-grp-service-service1] service tcp destination eq 21
[Device-obj-grp-service-service1] quit
[Device] object-group service service2
[Device-obj-grp-service-service2] service tcp destination eq 80
[Device-obj-grp-service-service2] quit
[Device] object-group service service3
[Device-obj-grp-service-service3] service tcp destination eq 8080
[Device-obj-grp-service-service3] quit
[Device] object-group service service4
[Device-obj-grp-service-service4] service tcp destination eq 25
[Device-obj-grp-service-service4] quit
```

# Configure global NAT rules to allow external users to access the internal servers.

```
[Device] nat global-policy
[Device-nat-global-policy] rule name rule1
[Device-nat-global-policy-rule-rule1] destination-ip host 202.38.1.1
[Device-nat-global-policy-rule-rule1] source-zone untrust
[Device-nat-global-policy-rule-rule1] service service1
[Device-nat-global-policy-rule-rule1] action dnat ip-address 10.110.10.3 local-port
21
[Device-nat-global-policy-rule-rule1] quit
[Device-nat-global-policy] rule name rule2
[Device-nat-global-policy-rule-rule2] destination-ip host 202.38.1.1
[Device-nat-global-policy-rule-rule2] source-zone untrust
[Device-nat-global-policy-rule-rule2] service service2
[Device-nat-global-policy-rule-rule2] action dnat ip-address 10.110.10.1 local-port
80
[Device-nat-global-policy-rule-rule2] quit
[Device-nat-global-policy] rule name rule3
[Device-nat-global-policy-rule-rule3] destination-ip host 202.38.1.1
[Device-nat-global-policy-rule-rule3] source-zone untrust
[Device-nat-global-policy-rule-rule3] service service3
```

```
[Device-nat-global-policy-rule-rule3] action dnat ip-address 10.110.10.2 local-port
80
[Device-nat-global-policy-rule-rule3] quit
[Device-nat-global-policy] rule name rule4
[Device-nat-global-policy-rule-rule4] destination-ip host 202.38.1.1
[Device-nat-global-policy-rule-rule4] source-zone untrust
[Device-nat-global-policy-rule-rule4] service service4
[Device-nat-global-policy-rule-rule4] action dnat ip-address 10.110.10.4 local-port
25
[Device-nat-global-policy-rule-rule4] quit
[Device-nat-global-policy] quit
```

Verifying the configuration

# Verify that the host on the external network can access the internal servers by using the public addresses. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT global-policy information:
  Totally 4 NAT global-policy rules.
  Rule name: rule1
    Type                  : nat
    DestIP address        : 202.38.1.1
    Source-zone name      : untrust
    Service object group  : service1
  DNAT action:
      IPv4 address: 10.110.10.3
      Port: 21
    NAT counting : 0
    Config status: Active

  Rule name: rule2
    Type                  : nat
    DestIP address        : 202.38.1.1
    Source-zone name      : untrust
    Destination-zone name : trust
    Service object group  : service2
  DNAT action:
      IPv4 address: 10.110.10.1
      Port: 80
    NAT counting : 0
    Config status: Active

  Rule name: rule3
    Type                  : nat
    DestIP address        : 202.38.1.1
    Source-zone name      : untrust
    Destination-zone name : trust
    Service object group  : service3
  DNAT action:
      IPv4 address: 10.110.10.2
      Port: 80
    NAT counting : 0
    Config status: Active

  Rule name: rule4
    Type                  : nat
    DestIP address        : 202.38.1.1
    Source-zone name      : untrust
    Destination-zone name : trust
    Service object group  : service4
  DNAT action:
      IPv4 address: 10.110.10.4
      Port: 25
    NAT counting : 0
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host accesses the FTP server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 202.38.1.2/52802
  Destination IP/port: 202.38.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.3/21
  Destination IP/port: 202.38.1.2/52802
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-21 11:13:39  TTL: 3597s
Initiator->Responder:            7 packets        313 bytes
Responder->Initiator:            6 packets        330 bytes
Total sessions found: 1
```

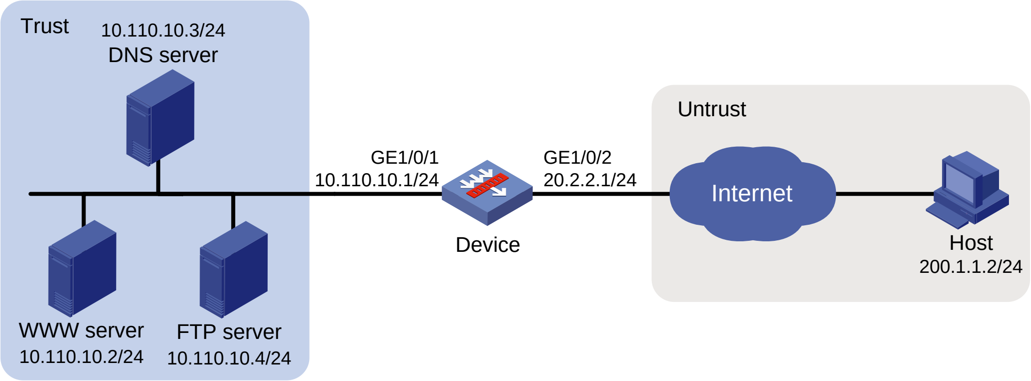
Example: Configuring NAT Server for external-to-internal access through domain name (non-overlapping addresses)

Network configuration

As shown in Figure 14, Web server at 10.110.10.2/24 in the internal network provides services for external users. A DNS server at 10.110.10.3/24 is used to resolve the domain name of the Web server. The company has two public IP addresses: 202.38.1.2 and 202.38.1.3.

Configure NAT Server to allow external users to access the internal Web server by using the domain name.

Figure 14 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure a NAT server mapping to map the private IP address and port of the DNS server to a public address and port. The mapping allows the external host to access the internal DNS server for domain name resolution.
- Enable ALG for DNS and configure outbound dynamic NAT to translate the private IP address of the Web server in the payload of the DNS response packet into a public IP address.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **untrust-trust** to permit the packets from the host to the servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.2
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.3
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.4
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure NAT:

# Enable NAT with ALG for DNS.

```
[Device] nat alg dns
```

# Create an address group.

```
[Device] nat address-group 1
[Device-address-group-1] address 202.38.1.3 202.38.1.3
[Device-address-group-1] quit
```

# Create service object group service1 for DNS service.

```
[Device] object-group service service1
[Device-obj-grp-service-service1] service tcp destination eq 53
[Device-obj-grp-service-service1] service udp destination eq 53
[Device-obj-grp-service-service1] quit
```

# Configure a global NAT rule to map the address 202.38.1.1 to 10.110.10.3. External users can access the internal DNS server.

```
[Device] nat global-policy
[Device-nat-global-policy] rule name rule1
[Device-nat-global-policy-rule-rule1] source-zone untrust
[Device-nat-global-policy-rule-rule1] destination-ip host 202.38.1.2
[Device-nat-global-policy-rule-rule1] service service1
[Device-nat-global-policy-rule-rule1] action dnat ip-address 10.110.10.3 local-port
53
[Device-nat-global-policy-rule-rule1] quit
```

# Configure a global NAT rule. Use the address in address group 1 to translate the private address in DNS response payload, and allow reversible NAT.

```
[Device-nat-global-policy] rule name rule2
[Device-nat-global-policy-rule-rule2] source-ip host 10.110.10.2
[Device-nat-global-policy-rule-rule2] source-zone trust
[Device-nat-global-policy-rule-rule2] destination-zone untrust
[Device-nat-global-policy-rule-rule2] action snat address-group 1 no-pat reversible
[Device-nat-global-policy-rule-rule2] quit
[Device-nat-global-policy] quit
```

Verifying the configuration

# Verify that the host on the external network can access the internal Web server by using the server's domain name. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 1
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.3            202.38.1.3

NAT global-policy information:
  Totally 2 NAT global-policy rules.
  Rule name: rule1
    Type                  : nat
    DestIP address        : 202.38.1.2
    Source-zone name      : untrust
    Service object group  : service1
  DNAT action:
      IPv4 address: 10.110.10.3
      Port: 53
    NAT counting : 0
    Config status: Active

  Rule name: rule2
    Type                  : nat
    SrcIP address         : 10.110.10.2
    Source-zone name      : trust
    Destination-zone name : untrust
  SNAT action:
      Address group ID: 1
      NO-PAT: Y
      Reversible: Y
      Port-preserved: N
    NAT counting : 0
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host accesses Web server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 200.1.1.2/1694
  Destination IP/port: 202.38.1.3/8080
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.2/8080
  Destination IP/port: 200.1.1.2/1694
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: -/-/-
Rule name:
Start time: 2017-06-15 14:53:29  TTL: 3597s
Initiator->Responder:           7 packets        308 bytes
Responder->Initiator:           5 packets        312 bytes
Total sessions found: 1
```

Example: Configuring NAT hairpin in P2P mode

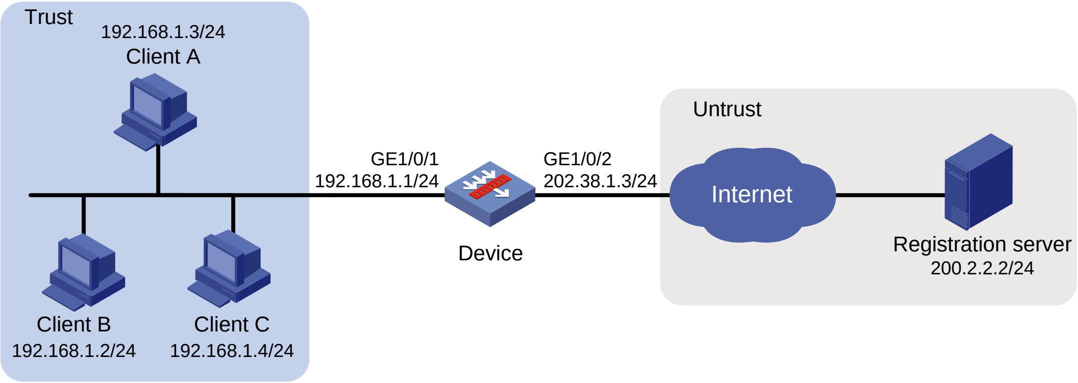
Network configuration

In the P2P application, internal clients must register their IP address to the external server and the server records the registered IP addresses and port numbers of the internal clients. An internal client must request the IP address and port number of another client from the external server before accessing the client.

Configure NAT hairpin so that:

- The internal clients can register the same public address to the external server.
- The internal clients can access each other through the IP address and port number obtained from the server.

Figure 15 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure outbound dynamic PAT on the NAT device, so the internal clients can access the external server for registration.
- Configure the mapping behavior for PAT as Endpoint-Independent Mapping because the registered IP address and port number should be accessible for any source address.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.1.

   ```
   [Device] ip route-static 200.2.2.0 24 202.38.1.1
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the clients to the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
   ```

```
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.2.2.2
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **trust-trust** to permit the packets between the clients in the **Trust** security zone.

```
[Device-security-policy-ip] rule name trust-trust
[Device-security-policy-ip-2-trust-trust] source-zone trust
[Device-security-policy-ip-2-trust-trust] destination-zone trust
[Device-security-policy-ip-2-trust-trust] source-ip-host 202.38.1.3
[Device-security-policy-ip-2-trust-trust] destination-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-2-trust-trust] action pass
[Device-security-policy-ip-2-trust-trust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

# Configure address object group **obj1** to identify packets from subnet 192.168.1.0/24.

```
[Device] object-group ip address obj1
[Device-obj-grp-ip-obj1] network subnet 192.168.1.0 24
[Device-obj-grp-ip-obj1] quit
```

# Configure a NAT rule for the global policy with Easy IP. IP address 202.38.1.1 is used as the public address for the source address translation of the packets from internal to external.

```
[Device] nat global-policy
[Device-nat-global-policy] rule name rule1
[Device-nat-global-policy-rule-rule1] source-zone trust
[Device-nat-global-policy-rule-rule1] destination-zone untrust
[Device-nat-global-policy-rule-rule1] source-ip obj1
[Device-nat-global-policy-rule-rule1] action snat easy-ip
[Device-nat-global-policy-rule-rule1] quit
[Device-nat-global-policy] quit
```

# Configure ACL 2000 to identify packets from subnet 192.168.1.0/24 to be translated.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

# Configure the Endpoint-Independent Mapping mode for PAT. For packets with the same source address and port number and permitted by ACL 2000, the source address and port number are translated to the same public address and port number.

```
[Device] nat mapping-behavior endpoint-independent acl 2000
```

Verifying the configuration

# Verify that Host A, Host B, and Host C can access each other after they register their IP addresses and port numbers to the external server. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT global-policy information:
  Totally 2 NAT global-policy rules.
  Rule name: rule1
    Type                  : nat
    SrcIP object group    : obj1
    Source-zone name      : trust
    Destination-zone name : untrust
  SNAT action:
      Easy-IP
      Reversible: N
      Port-preserved: N
    NAT counting : 0
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Endpoint-Independent
  ACL          : 2000
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Client A accesses Client B.

```
[Device] display nat session verbose
Slot 1:
```

```
Initiator:
  Source      IP/port: 192.168.1.3/44929
  Destination IP/port: 202.38.1.3/1
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: UDP(17)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 192.168.1.2/69
  Destination IP/port: 202.38.1.3/1024
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: UDP(17)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: UDP_READY
Application: TFTP
Rule ID: -/-/-
Rule name:
Start time: 2012-08-15 15:53:36  TTL: 46s
Initiator->Responder:           1 packets         56 bytes
Responder->Initiator:           1 packets         72 bytes
Total sessions found: 1
```

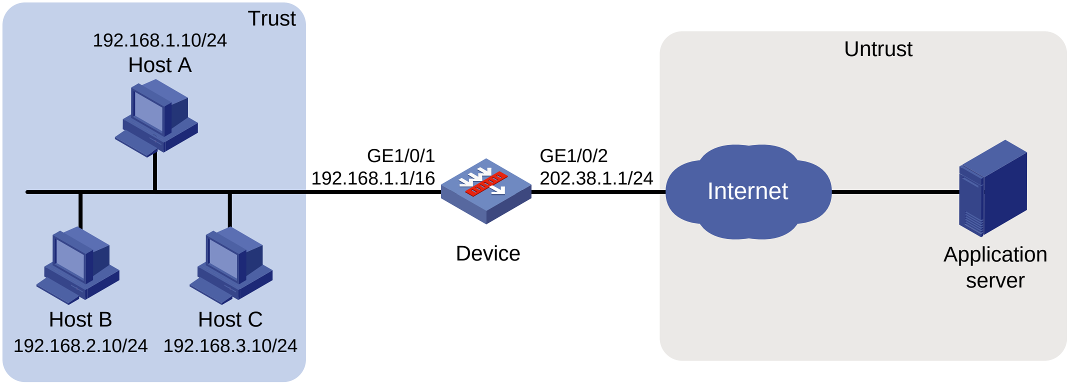
Example: Configuring the global NAT policy for NAT444 dynamic port mapping

Network configuration

As shown in Figure 16, a company uses private IP address on network 192.168.0.0/16 and public IP addresses 202.38.1.2 and 202.38.1.3. Configure the global NAT policy to meet the following requirements:

- Only users on subnet 192.168.1.0/24 can use public IP addresses 202.38.1.2 and 202.38.1.3 to access the server at 200.2.2.1 on the Internet.
- The port range for the public IP addresses is 1024 to 65535.
- The port block size is 300.
- If the ports in the assigned port block are all used, extend another port block for users.

Figure 16 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 16
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.20.

```
[Device] ip route-static 200.2.2.1 32 202.38.1.20
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the hosts to the application server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.2.2.1
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

   # Create NAT address group **0**.

```
[Device] nat address-group 0
```

   # Add public IP addresses 202.38.1.2 and 202.38.1.3 to the NAT address group.

```
[Device-address-group-0] address 202.38.1.2 202.38.1.3
```

# Configure the port range as 1024 to 65535.

```
[Device-address-group-0] port-range 1024 65535
```

# Set the port block size to 300 and the extended port block number to 1.

```
[Device-address-group-0] port-block block-size 300 extended-block-number 1
[Device-address-group-0] quit
```

# Configure address object group **obj1** to identify packets from subnet 192.168.1.0/24.

```
[Device] object-group ip address obj1
[Device-obj-grp-ip-obj1] network subnet 192.168.1.0 24
[Device-obj-grp-ip-obj1] quit
```

# Configure a NAT rule for the global policy, and specify address object group **obj1** as the packet match criterion and use NAT address group **0** for source address and port translation.

```
[Device] nat global-policy
[Device-nat-global-policy] rule name rule1
[Device-nat-global-policy-rule-rule1] source-ip obj1
[Device-nat-global-policy-rule-rule1] action snat address-group 0
```

Verifying the configuration

# Verify that Host A can access external servers, but Host B and Host C cannot. (Details not shown.)

# Display all NAT configurations and statistics.

```
[Device]display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 0
    Port range: 1024-65535
Blade-load-sharing-group: Blade4fw-m90001
    Port block size: 300
    Extended block number: 1
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.3
    Exclude address information:
      Start address         End address

NAT global-policy information:
  Totally 1 NAT global-policy rules.
  Rule name: rule1
    SrcIP object group    : obj1
  SNAT action:
      Address group ID: 0
      NO-PAT: N
      Reversible: N
      Port-preserved: N
    NAT counting : 0
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing: Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT statistics.

```
[Device] display nat statistics
Slot 1:
  Total session entries: 1
  Session creation rate: 0
  Total EIM entries: 0
  Total inbound NO-PAT entries: 0
  Total outbound NO-PAT entries: 0
  Total static port block entries: 0
  Total dynamic port block entries: 430
  Active static port block entries: 0
  Active dynamic port block entries: 1
```

Display dynamic port block entries.

```
[Device] display nat port-block dynamic
Slot 1:
Local VPN     Local IP         Global IP        Port block   Connections
---           192.168.1.10     202.38.1.2       45724-46023  1
Total mappings found: 1
```

Example: Configuring a hot backup system in active/standby mode in collaboration with VRRP for NAT

For more information, see *High Availability Configuration Guide*.

Example: Configuring a hot backup system in dual-active mode in collaboration with VRRP for NAT

For more information, see *High Availability Configuration Guide*.

Configuring interface-based NAT

Restrictions: Hardware compatibility with NAT configuration

Interface-based NAT is not supported on the NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280 devices.

Restrictions and guidelines: interface-based NAT configuration

The general restrictions and guidelines are as follows:

- You can use an ACL in a NAT rule to identify the IP addresses to be translated. The match criteria include the source IP address, source port number, destination IP address, destination port number, transport layer protocol, and VPN instance.
- If NAT is configured on only one output interface in a dual uplink network, do not add the two output interfaces to the same security zone. Doing so will cause communication interruption. For more information about security zone, see *Security Configuration Guide.*
- As a best practice, configure interface-based NAT on a logical interface of the device with multiple slots so that all packets to be NATed can be processed on the logical interface. If you fail to do so, the slots for input and output packets might be different and address translation will fail.
- If you perform all the translation methods on an interface, the NAT rules are sorted in the following descending order:
  a. NAT Server.
  b. Static NAT.
  c. NAT444 static port blocking mapping.
  d. Dynamic NAT, NAT444 dynamic port block mapping, and DS-Lite B4 address translation.
  Dynamic NAT, NAT444 dynamic port block mapping, and DS-Lite B4 address translation have the same priority. Dynamic NAT rules and NAT444 dynamic port block mapping rules are sorted in descending order of ACL numbers and are effective for IPv4 packets. DS-Lite B4 address translation rules are effective for IPv6 packets.

When you configure BRAS unification, follow these restrictions and guidelines:

- Supported user address types are private IPv4 address, private-DS address, and DS-Lite address.
- If the NAT444 configuration changes after users get online, the public IP addresses and port numbers used by the users also change. The change cannot be synchronized to the AAA server, affecting user tracing accuracy. As a best practice, log off the users immediately after you change the NAT444 configuration. When the users come online again, NAT444 creates new mappings for them.

Interface-based NAT tasks at a glance

To configure NAT, perform the following tasks:

1. Configuring an address translation method on an interface

    - Configuring static NAT on an interface
    - Configuring dynamic NAT on an interface
    - Configuring NAT server mappings on an interface
    - Configuring NAT444 on an interface
    - Configuring DS-Lite B4 address translation on an interface
    - Configuring the interface-based NAT policy

      The NAT policy allows flexible address translation rules for multiple interfaces.
2. (Optional.) Configuring NAT hairpin
3. (Optional.) Configuring NAT ALG
4. (Optional.) Configuring NAT DNS mapping
5. (Optional.) Configuring high availability for NAT
    - Enabling NAT port halving
    - Configuring hot backup for NAT
6. (Optional.) Configuring NAT maintenance
    - Configuring periodic NAT statistics collection
    - Enabling statistics collection for NAT session creation rate
    - Specifying a probe method for detecting reachability of NAT address group members
    - Enabling sending ICMP error messages for NAT failures
    - Enabling NAT configuration changes to take effect only on new connections
7. (Optional.) Configuring NAT logging
8. (Optional.) Configuring NAT in specific networks
    - Enabling NAT reply redirection
    - Enabling the deletion of timestamps in TCP SYN and SYN ACK packets
    - Enabling NAT session recreation after link switchover

Configuring static NAT on an interface

Restrictions and guidelines for static NAT configuration on an interface

Typically, configure inbound static NAT with outbound dynamic NAT, NAT Server, or outbound static NAT to implement source address translation and destination address translation.

Prerequisites

Before configuring static NAT, you must perform the following tasks:

- Configure an ACL to identify the IP addresses to be translated. For more information about ACLs, see *ACL and QoS Configuration Guide*.
- Manually add a route for inbound static NAT. Use *local-ip* or *local-network* as the destination address, and use *global-ip*, an address in *global-network*, or the next hop directly connected to the output interface as the next hop.

Configuring outbound one-to-one static NAT

About this task

For address translation from a private IP address to a public IP address, configure outbound one-to-one static NAT on the interface connected to the external network.

- When the source IP address of an outgoing packet matches the *local-ip*, the source IP address is translated into the *global-ip*.
- When the destination IP address of an incoming packet matches the *global-ip*, the destination IP address is translated into the *local-ip*.

Restrictions and guidelines

If multiple outbound one-to-one static mappings uses different ACL rules, you can map a private address to different public addresses.

If you do not specify the **acl** keyword for a one-to-one mapping NAT rule, the rule can process reversible traffic. If the **acl** keyword is specified for the NAT rule, you must specify the **reversible** keyword for the rule to process reversible traffic.

Procedure

1. Enter system view.

   system-view

2. Configure a one-to-one mapping for outbound static NAT.

   nat static outbound *local-ip* [**vpn-instance** *local-vpn-instance-name*] *global-ip* [**vpn-instance** *global-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } [**reversible**]] [**vrrp** *virtual-router-id*] [**rule** *rule-name*] [**priority** *priority*] [**disable**] [**counting**] [**description** *text*]

3. (Optional.) Rearrange outbound one-to-one mapping NAT rules to adjust their priorities.

   nat static outbound rule move *nat-rule-name1* { **after** | **before** } *nat-rule-name2*

4. Enter interface view.

   interface *interface-type interface-number*

5. Enable static NAT on the interface.

   nat static enable

   By default, static NAT is disabled.

Configuring outbound net-to-net static NAT

About this task

For address translation from a private network to a public network, configure outbound net-to-net static NAT on the interface connected to the external network.

- When the source IP address of an outgoing packet matches the private address range, the source IP address is translated into a public address in the public address range.
- When the destination IP address of an incoming packet matches the public address range, the destination IP address is translated into a private address in the private address range.

Procedure

1. Enter system view.

   system-view

2. Configure a net-to-net mapping for outbound static NAT.

   ```
   nat static outbound net-to-net local-start-address local-end-address
   [ vpn-instance local-vpn-instance-name ] global global-network
   { mask-length | mask } [ vpn-instance global-vpn-instance-name ] [ acl
   { ipv4-acl-number | name ipv4-acl-name } [ reversible ] ] [ vrrp
   virtual-router-id ] [ rule rule-name ] [ priority priority ] [ disable ]
   [ counting ]
   ```

3. (Optional.) Change the priority of the outbound net-to-net static NAT rule.

   ```
   nat static outbound net-to-net rule move nat-rule-name1 { after |
   before } nat-rule-name2
   ```

   By default, an outbound net-to-net static NAT rule appearing earlier on the rule list has a higher priority for packet matching.

4. Enter interface view.

   ```
   interface interface-type interface-number
   ```

5. Enable static NAT on the interface.

   ```
   nat static enable
   ```

   By default, static NAT is disabled.

Configuring object group-based outbound static NAT

About this task

Configure object group-based outbound static NAT on the interface connected to the external network to translate private IP addresses into public IP addresses.

- When the source address of an outgoing packet matches the private address object group, the source address is translated into a public address in the public address object group.
- When the destination address of an incoming packet matches the public address object group, the destination address is translated into a private address in the private address object group.

Restrictions and guidelines

If you specify the **acl** keyword, NAT processes only packets permitted by the ACL.

For an object group-based outbound static NAT mapping to take effect, make sure the following requirements are met:

- One address object group contains only one host object or subnet object.
- A subnet object cannot have excluded addresses.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure an object group-based outbound static NAT mapping.

   ```
   nat static outbound object-group local-object-group-name
   [ vpn-instance local-vpn-instance-name ] object-group
   global-object-group-name [ vpn-instance global-vpn-instance-name ]
   [ acl { ipv4-acl-number | name ipv4-acl-name } [ reversible ] ] [ vrrp
   virtual-router-id ] [ disable ] [ counting ]
   ```

   By default, no mappings exist.

3. Enter interface view.

   ```
   interface interface-type interface-number
   ```

4. Enable static NAT on the interface.

```
nat static enable
```

By default, static NAT is disabled.

Configuring inbound one-to-one static NAT

About this task

For address translation from a public IP address to a private IP address, configure inbound one-to-one static NAT.

- When the source IP address of an incoming packet matches the *global-ip*, the source IP address is translated into the *local-ip*.
- When the destination IP address of an outgoing packet matches the *local-ip*, the destination IP address is translated into the *global-ip*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure a one-to-one mapping for inbound static NAT.

   ```
   nat static inbound global-ip [ vpn-instance global-vpn-instance-name ]
   local-ip [ vpn-instance local-vpn-instance-name ] [ acl
   { ipv4-acl-number | name ipv4-acl-name } [ reversible ] ] [ rule
   rule-name ] [ priority priority ] [ disable ] [ counting ] [ description
   text ]
   ```

3. Enter interface view.

   ```
   interface interface-type interface-number
   ```

4. Enable static NAT on the interface.

   ```
   nat static enable
   ```

   By default, static NAT is disabled.

Configuring inbound net-to-net static NAT

About this task

For address translation from a public network to a private network, configure inbound net-to-net static NAT.

- When the source IP address of an incoming packet matches the public address range, the source IP address is translated into a private address in the private address range.
- When the destination IP address of an outgoing packet matches the private address range, the destination IP address is translated into a public address in the public address range.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure a net-to-net mapping for inbound static NAT.

   ```
   nat static inbound net-to-net global-start-address
   global-end-address [ vpn-instance global-vpn-instance-name ] local
   local-network { mask-length | mask } [ vpn-instance
   local-vpn-instance-name ] [ acl { ipv4-acl-number | name ipv4-acl-name }
   [ reversible ] ] [ rule rule-name ] [ priority priority ] [ disable ]
   [ counting ]
   ```

3. (Optional.) Change the priority of the inbound net-to-net static NAT rule.

nat static inbound net-to-net rule move *nat-rule-name1* { **after** | **before** } *nat-rule-name2*

By default, an inbound net-to-net static NAT rule appearing earlier on the rule list has a higher priority for packet matching.

4. Enter interface view.

   interface *interface-type interface-number*

5. Enable static NAT on the interface.

   nat static enable

   By default, static NAT is disabled.

Configuring object group-based inbound static NAT

About this task

Configure object group-based inbound static NAT to translate public IP addresses into private IP addresses.

- When the destination address of an outgoing packet matches the private address object group, the destination address is translated into a public address in the public address object group.
- When the source address of an incoming packet matches the public address object group, the source address is translated into a private address in the private address object group.

Restrictions and guidelines

If you specify the **acl** keyword, NAT processes only packets permitted by the ACL.

For an object group-based inbound static NAT mapping to take effect, make sure the following requirements are met:

- One address object group contains only one host object or subnet object.
- A subnet object cannot have excluded addresses.

For an inbound static mapping, if its private IPv4 object group contains a host address, the host address cannot be on the same subnet as the interface configured with this mapping.

Procedure

1. Enter system view

   system-view

2. Configure an object group-based inbound static NAT mapping.

   nat static inbound object-group *global-object-group-name* [**vpn-instance** *global-vpn-instance-name*] **object-group** *local-object-group-name* [**vpn-instance** *local-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } [**reversible**]] [**disable**] [**counting**]

   By default, no NAT mappings exist.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable static NAT on the interface.

   nat static enable

   By default, static NAT is disabled.

Configuring dynamic NAT on an interface

Restrictions and guidelines dynamic NAT configuration on an interface

You can configure multiple inbound or outbound dynamic NAT rules.

- A NAT rule with an ACL takes precedence over a rule without any ACL.
- If two ACL-based dynamic NAT rules are configured, the rule with the higher ACL number has higher priority.

When configuring NO-PAT and DNS ALG for internal server access through a domain name, make sure the NAT address group for NO-PAT has sufficient IP addresses. The minimum number of IP addresses must be equal to the number of security engines multiplied by the number of internal servers. For more information about security engines, see context configuration in *Virtual Technologies Configuration Guide*.

Prerequisites

Before configuring dynamic NAT, you must perform the following tasks:

- Configure an ACL to identify the IP addresses to be translated. For more information about ACLs, see *ACL and QoS Configuration Guide*.
- Determine whether to enable the Easy IP feature. If you use the IP address of an interface as the NAT address, you are configuring Easy IP.
- Determine a public IP address pool for address translation.
- Determine whether to translate port numbers. Use NO-PAT to translate only IP addresses and PAT to translate both IP addresses and port numbers.

Configuring outbound dynamic NAT

About this task

To translate private IP addresses into public IP addresses, configure outbound dynamic NAT on the interface connected to the external network.

Procedure

1. Enter system view.

   system-view

2. Create a NAT address group and enter its view.

   nat address-group *group-id*

3. Add an address range to the address group.

   address *start-address end-address* [**name** *group-name*]

   You can add multiple address ranges to an address group.

   The address ranges must not overlap.

4. (Optional.) Exclude IP addresses from being used in address translation.

   exclude-ip *start-address end-address*

   The *end-address* must not be lower than the *start-address*. If they are the same, you specify only one IP address.

5. Return to system view.

quit

6. Enter interface view.

   interface *interface-type interface-number*

7. Configure outbound dynamic NAT. Choose the options to configure as needed:

   - Configure NO-PAT.

     nat outbound [*ipv4-acl-number* | **name** *ipv4-acl-name*] **address-group** { *group-id* | **name** *group-name* } [**vpn-instance** *vpn-instance-name*] **no-pat** [**reversible**] [**rule** *rule-name*] [**priority** *priority*] [**disable**] [**counting**] [**description** *text*]

   - Configure PAT.

     nat outbound [*ipv4-acl-number* | **name** *ipv4-acl-name*][**address-group** { *group-id* | **name** *group-name* }] [**vpn-instance** *vpn-instance-name*] [**port-preserved**] [**rule** *rule-name*] [**priority** *priority*] [**disable**] [**counting**] [**description** *text*]

   You can configure multiple outbound dynamic NAT rules on an interface.

Parameter	Description
address-group	If you do not specify this keyword, the IP address of the interface is used as the NAT address. Easy IP is implemented.
no-pat reversible	If you specify these keywords, you enable reverse address translation. Reverse address translation uses existing NO-PAT entries to translate the destination address for connections actively initiated from the external network to the internal network. The destination address is translated into the private IP address in the matching NO-PAT entry.

8. (Optional.) Configure a PAT mapping mode.

   a. Return to system view.

      quit

   b. Configure a PAT mapping mode.

      nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

      The default mapping mode is **Address and Port-Dependent Mapping**.

      This command takes effect only on outbound dynamic NAT for PAT.

9. (Optional.) Rearrange outbound dynamic NAT rules to adjust their priorities.

   nat outbound rule move *nat-rule-name1* { **after** | **before** } *nat-rule-name2*

Configuring inbound dynamic NAT

Restrictions and guidelines

Do not configure inbound dynamic NAT alone. Typically, inbound dynamic NAT functions with outbound dynamic NAT, NAT Server, or outbound static NAT to implement source address translation and destination address translation.

As a best practice, manually create a route because it takes time to automatically add routes.

Procedure

1. Enter system view.

   system-view

2. Create a NAT address group and enter its view.

nat address-group *group-id*

3. Add an address range to the address group.

   address *start-address end-address* [**name** *group-name*]

   You can add multiple address ranges to an address group.

   The address ranges in address groups must not overlap.

4. (Optional.) Exclude IP addresses from being used in address translation.

   exclude-ip *start-address end-address*

   The *end-address* must not be lower than the *start-address*. If they are the same, you specify only one IP address.

5. Return to system view.

   quit

6. Enter interface view.

   interface *interface-type interface-number*

7. Configure inbound dynamic NAT.

   nat inbound { *ipv4-acl-number* | **name** *ipv4-acl-name* } **address-group** { *group-id* | **name** *group-name* } [**vpn-instance** *vpn-instance-name*] [**no-pat** [**reversible**] [**add-route**]] [**rule** *rule-name*] [**priority** *priority*] [**disable**] [**counting**] [**description** *text*]

   You can configure multiple inbound dynamic NAT rules on an interface.

Parameter	Description
no-pat reversible	If you specify these keywords, you enable reverse address translation. Reverse address translation uses existing NO-PAT entries to translate the destination address for connections actively initiated from the external network to the internal network. The destination address is translated into the private IP address in the matching NO-PAT entry.
add-route	This keyword enables the device to automatically add a route destined for the private address when an inbound dynamic NAT rule is matched. The output interface is the NAT interface, and the next hop is the source address before translation. If you do not specify this keyword, you must manually add the route.

8. (Optional.) Rearrange inbound dynamic NAT rules to adjust their priorities.

   nat inbound rule move *nat-rule-name1* { **after** | **before** } *nat-rule-name2*

Configuring NAT server mappings on an interface

About NAT server mappings

Typically, the NAT Server feature is configured on the interface connected to the external network to allow servers in the internal network or an MPLS VPN instance to provide services for external users. A NAT server mapping (also called NAT server rule) maps a public IP address and port number to the private IP address and port number of the internal server.

The NAT Server feature can be implemented by configuring the following server mappings:

- **Common NAT server mapping**—Maps the private IP address and the port number of the internal server to a public IP address and a port number. This method allows external hosts to access the internal server by using the specified public IP address.
- **Load sharing NAT server mapping**—You can add multiple internal servers to an internal server group so that these servers provide the same service for external hosts. The NAT device

chooses one internal server based on the weight and number of connections of the servers to respond to a request from an external host to the public address of the internal server group.

- **ACL-based NAT server mapping**—An extension of common NAT server mapping. A common NAT server mapping maps the private IP address of the internal server to a single public IP address. An ACL-based NAT server mapping maps the private IP address of the internal server to a set of public IP addresses defined by an ACL. If the destination address of a packet matches a permit rule in the ACL, the destination address is translated into the private IP address of the internal server.
- **Object group-based server mapping**—Uses the address object group and service object group as the packet match criteria. The destination IP addresses and port numbers of the matching packets are translated to the same address and port number. For more information about object groups, see *Security Configuration Guide*.

Restrictions and guidelines for NAT server mapping configuration on an interface

In a scenario where Real-Time Transport Protocol (RTP) is used to transmit audio and video flows, the server and client send Real-Time Control Protocol (RTCP) packets periodically during the course of a session. If the server is on the private network, you must specify the reversible keyword when executing the nat server command. If you do not specify the reversible keyword, the NAT device discards the RTCP packets sent to the client on the public network and audio and video transmission services fail.

When you configure a load shared NAT server mapping, you must make sure a user uses the same public address and public port to access the same service on an internal server. For this purpose, make sure value *N* in the following mappings is equal to or less than the number of servers in the internal server group:

- One public address and *N* consecutive public port numbers are mapped to one internal server group.
- *N* consecutive public addresses and one public port number are mapped to one internal server group.

When you roll back configuration in a version that supports the automatic NAT rule name assignment, a rollback failure message is displayed if the no automatically assigned names exist in the replacement configuration file.

For example, the system compares the configuration in the replacement configuration file and the configuration after the rollback and displays a rollback failure message in the following conditions:

- The replacement configuration file has the following configuration: **nat server global** 112.1.1.1 **inside** 192.168.20.1.
- The NAT rule configuration after the rollback is **nat server global** 112.1.1.1 **inside** 192.168.20.1 rule NAT server rule_10 (NAT server rule_10 indicating an NAT rule name automatically assigned by the system).

In this case, the NAT rule configuration in the replacement configuration file has been issued and you can ignore this failure message.

Configuring common NAT server mappings

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Configure common NAT server mappings. Choose the options to configure as needed:

- A single public address with a single or no public port:

  nat server [**protocol** *pro-type*] **global**{ *global-address* | **current-interface** | **interface** *interface-type interface-number* } [*global-port*] [**vpn-instance** *global-vpn-instance-name*] **inside** *local-address* [*local-port*] [**vpn-instance** *local-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }] [**reversible**] [**vrrp** *virtual-router-id*] [**rule** *rule-name*] [**disable**] [**counting**] [**description** *text*]

- A single public address with consecutive public ports:

  nat server protocol *pro-type* **global**{ *global-address* | **current-interface** | **interface** *interface-type interface-number* } *global-port1 global-port2* [**vpn-instance** *global-vpn-instance-name*] **inside** { { *local-address* | *local-address1 local-address2* } *local-port* | *local-address local-port1 local-port2* } [**vpn-instance** *local-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }] [**vrrp** *virtual-router-id*] [**rule** *rule-name*] [**disable**] [**counting**] [**description** *text*]

- Consecutive public addresses with no public port:

  nat server protocol *pro-type* **global** *global-address1 global-address2* [**vpn-instance** *global-vpn-instance-name*] **inside** { *local-address* | *local-address1 local-address2* } [*local-port*] [**vpn-instance** *local-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }] [**vrrp** *virtual-router-id*] [**rule** *rule-name*] [**disable**] [**counting**] [**description** *text*]

- Consecutive public addresses with a single public port:

  nat server protocol *pro-type* **global** *global-address1 global-address2 global-port* [**vpn-instance** *global-vpn-instance-name*] **inside** { *local-address* [*local-port1 local-port2*] | [*local-address* | *local-address1 local-address2*] [*local-port*] } [**vpn-instance** *local-vpn-instance-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }] [**vrrp** *virtual-router-id*] [**rule** *rule-name*] [**disable**] [**counting**] [**description** *text*]

You can configure multiple NAT server mappings on an interface.

Configuring load sharing NAT server mappings

1. Enter system view.

   system-view

2. Create a NAT server group and enter its view.

   nat server-group *group-id*

   By default, no NAT server groups exist.

3. Add an internal server into the group.

   inside ip *inside-ip* **port** *port-number* [**weight** *weight-value*]

   You can add multiple internal servers to a group.

4. Return to system view.

   quit

5. Enter interface view.

   interface *interface-type interface-number*

6. Configure a load sharing NAT server mapping.

```
nat server protocol pro-type global { { global-address | nat server
protocol pro-type global { { global-address | current-interface |
interface interface-type interface-number } { global-port |
global-port1 global-port2 } | global-address1 global-address2
global-port } [ vpn-instance global-vpn-instance-name ] inside
server-group group-id[ vpn-instance local-vpn-instance-name ] [ acl
{ ipv4-acl-number | name ipv4-acl-name } ] [ vrrp virtual-router-id ]
[ rule rule-name ] [ disable ] [ counting ] [ description text ]
```

You can configure multiple load sharing NAT server mappings on an interface.

Configuring ACL-based NAT server mappings

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Configure an ACL-based NAT server mapping.

```
nat server global { ipv4-acl-number | name ipv4-acl-name } inside
local-address [ local-port ] [ vpn-instance local-vpn-instance-name ]
[ vrrp virtual-router-id ] [ rule rule-name ] [ priority priority ]
[ disable ] [ counting ] [ description text ]
```

You can configure multiple NAT server mappings on an interface.

4. (Optional.) Rearrange ACL-based NAT server mappings to adjust their priorities.

```
nat server rule move nat-rule-name1 { after | before } nat-rule-name2
```

Configuring object group-based NAT server mappings

About this task

An object group-based NAT server mapping uses the address object group and service object group as the packet match criteria. The destination IP addresses and port numbers of the matching packets are translated to the same address and port number. For more information about object groups, see *Security Configuration Guide*.

When multiple object group-based NAT server mappings are configured, the mapping configured earlier has a higher priority. The match process of a packet stops when the packet matches a mapping.

Restrictions and guidelines

The private port number in the NAT server mapping takes effect only when the protocol type is TCP or UDP for the service object group used by the mapping.

One NAT server mapping supports a maximum of five address object groups and one service object group.

Prerequisites

Before you create a mapping, the IPv4 address object groups and service object group must already exist. An IPv4 address object group cannot have excluded IPv4 address configuration.

Procedure

1. Enter system view

```
system-view
```

2. Enter interface view.

interface *interface-type interface-number*

3. Create an object group-based NAT server mapping.

   nat server rule *rule-name* **global destination-ip** *object-group-name*&<1-5> [**service** *object-group-name*] **inside** *local-address* [*local-port*] [**vrrp** *virtual-router-id*] [**disable**] [**counting**] [**description** *text*]

   By default, no object group-based NAT server mapping exists.
4. Add object groups to the NAT server mapping.

   nat server rule *rule-name* **global** { **destination-ip** *object-group-name*&<1-5> | **service** *object-group-name* }

   You can add object groups only to existing object group-based NAT server mappings.

Configuring NAT444 on an interface

About NAT444

NAT444 provides outbound address translation, and it is configured on the interface connected to the public network. By configuring NAT444 address translation on the NAT444 gateway, multiple private IP addresses are mapped to one public IP address and a different port block is used for each private IP address

Restrictions and guidelines for NAT444 configuration on an interface

To configure dynamic port block mapping for NAT444, you must configure port block parameters in the NAT address group.

Configuring static port block mapping for NAT444

1. Enter system view.

   system-view
2. Create a NAT port block group, and enter its view.

   nat port-block-group *group-id*
3. Add a private IP address range to the port block group.

   local-ip-address *start-address end-address* [**vpn-instance** *vpn-instance-name*]

   You can add multiple private IP address ranges to one port block group, but they cannot overlap.
4. Add a public IP address range to the port block group.

   global-ip-pool *start-address end-address*

   You can add multiple public IP address ranges to one port block group, but they cannot overlap.
5. Configure the port range for the public IP addresses.

   port-range *start-port-number end-port-number*

   By default, the port range is 1 to 65535.
6. Set the port block size.

   block-size *block-size*

By default, the port block size is 256.

7. Return to system view.

   quit

8. Enter interface view.

   interface *interface-type interface-number*

9. Configure a static outbound port block mapping rule on the interface.

   nat outbound port-block-group *group-id* [**rule** *rule-name*] [**counting**]

   By default, no port block mapping rule is configured on an interface.

   You can configure multiple port block mapping rules on one interface.

10. (Optional.) Configure a PAT mapping mode.

    a. Return to system view.

       quit

    b. Configure a PAT mapping mode.

       nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

       The default mapping mode is **Address and Port-Dependent Mapping**.

Configuring dynamic port block mapping for NAT444

You can use one of the following methods to add IP addresses to a NAT address group:

- **Method 1**—Adding one or multiple address ranges.
- **Method 2**—Adding the IP address of the specified interface. This method supports the cooperation between dynamic NAT port block mappings and Easy IP and supports user tracing.

Only one method is supported for a NAT address group.

Use Method 2 if the IP address after translation is the IP address of the interface connected to the external network and the IP address is dynamically obtained through DHCP. This method avoids the inaccurate NAT IP address information caused by the IP address change of the interface.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure a PAT mapping mode.

   nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   The default mapping mode is **Address and Port-Dependent Mapping**.

3. Create a NAT address group, and enter its view.

   nat address-group *group-id* [**name** *group-name*]

4. Add addresses to the NAT address group. Choose one option as needed:

   ○ Add IP address ranges to the NAT address group.

   address *start-address end-address*

   You can add multiple public IP address ranges to an address group.

   The IP address ranges in address groups cannot overlap.

   Execute the following command to exclude IP addresses from being used in address translation.

   exclude-ip *start-address end-address*

The *end-address* must not be lower than the *start-address*. If they are the same, you specify only one IP address.

- Add the IP address of an interface to the NAT address group.

  address interface *interface-type interface-number*

  By default, no interface address exists in the NAT address group.

  You can specify only one interface for a NAT address group.

5. (Optional.) Configure the port range for the public IP addresses.

   port-range *start-port-number end-port-number*

   By default, the port range is 1 to 65535.

   The configuration takes effect only on PAT translation mode.

6. Configure port block parameters.

   port-block block-size *block-size* [**extended-block-number** *extended-block-number*]

   The configuration takes effect only on PAT translation mode.

7. Return to system view.

   quit

8. Enter interface view.

   interface *interface-type interface-number*

9. Configure PAT for outbound dynamic NAT.

   nat outbound [*ipv4-acl-number* | **name** *ipv4-acl-name*] [**address-group** { *group-id* | **name** *group-name* }] [**vpn-instance** *vpn-instance-name*] [**port-preserved**] [**rule** *rule-name*] [**priority** *priority*] [**disable**] [**counting**] [**description** *text*]

   By default, no outbound dynamic NAT rules exist.

   The **port-preserved** keyword does not take effect on dynamic NAT444.

10. (Optional.) Enable dynamic port block mapping synchronization.

    a. Return to system view.

       quit

    b. Enable dynamic port block mapping synchronization.

       nat port-block synchronization enable

       By default, dynamic port block mapping synchronization is disabled.

Enabling port block global sharing

About this task

When multiple interfaces have dynamic NAT port block mapping configured, the interfaces might create different port block mappings for packets from the same IP address. You can use this command to configure the interfaces to use the same port block mapping for translating packets from the same IP address.

Procedure

1. Enter system view

   system-view

2. Enable port block global sharing.

   nat port-block global-share enable

   By default, port block global sharing is disabled.

Configuring DS-Lite B4 address translation on an interface

About this task

DS-Lite B4 address translation is configured on the AFTR's interface connected to the external network. It uses an IPv6 ACL to identify packets to be NATed.

DS-Lite B4 address translation supports only the dynamic port block mapping method.

Prerequisites

Before configuring DS-Lite B4 address translation, make sure the B4 element and the AFTR can reach each other through IPv6.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure a PAT mapping mode.

   nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   The default mapping mode is **Address and Port-Dependent Mapping**.

3. Create a NAT address group, and enter its view.

   nat address-group *group-id* [**name** *group-name*]

4. Add a public IP address range to the NAT address group.

   address *start-address end-address*

   You can add multiple public IP address ranges to an address group.

   The IP address ranges in address groups cannot overlap.

5. (Optional.) Exclude IP addresses from being used in address translation.

   exclude-ip *start-address end-address*

   The *end-address* must not be lower than the *start-address*. If they are the same, you specify only one IP address.

6. Configure the port range for the public IP addresses.

   port-range *start-port-number end-port-number*

   By default, the port range is 1 to 65535.

   The configuration takes effect only on PAT translation mode.

7. Configure port block parameters.

   port-block block-size *block-size* [**extended-block-number** *extended-block-number*]

   By default, no port block parameters exist.

   The configuration takes effect only on PAT translation mode.

8. Return to system view.

   quit

9. Enter interface view.

   interface *interface-type interface-number*

10. Configure DS-Lite B4 address translation.

    nat outbound ds-lite-b4 { *ipv6-acl-number* | **name** *ipv6-acl-name* } **address-group** *group-id*

By default, DS-Lite B4 address translation is not configured.

11. (Optional.) Enable dynamic port block mapping synchronization.
    a. Return to system view.
       quit
    b. Enable dynamic port block mapping synchronization.
       nat port-block synchronization enable
       By default, dynamic port block mapping synchronization is disabled.

Configuring the interface-based NAT policy

About the interface-based NAT policy

The interface-based NAT policy performs address translation for outgoing packets on the interfaces that the rules are applied. The NAT policy can contain a set of NAT rules. The device identifies the packets based on the object groups in the NAT rules, and translates addresses according to the method in the matching rule.

Hardware compatibility with interface-based NAT policy

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

Restrictions and guidelines for interface-based NAT policy configuration

If a NAT rule does not use any object group, this rule matches all packets.

The NAT policy supports only dynamic address translation, and the policy has a higher priority than the dynamic address translation configuration on interfaces.

Creating the interface-based NAT policy

1. Enter system view
   system-view
2. Create the interface-based NAT policy and enter its view.
   nat policy
   By default, the interface-based NAT policy does not exist.

Configuring NAT rules

Restrictions and guidelines

The NAT rules in the interface-based NAT policy are sorted in descending order of their configuration order. A rule configured earlier has a higher priority. The matching process stops when a packet matches a NAT rule. You can use the **display this** command to view the configuration order of the NAT rules.

Procedure

1. Enter system view.

   system-view

2. Enter the interface-based NAT policy view.

   nat policy

3. Create a NAT rule and enter its view.

   rule name *rule-name*

   By default, no NAT rule exists.

4. (Optional.) Configure a description for the rule.

   description *text*

   By default, a NAT rule does not have any description.

5. Apply the NAT rule to the outgoing traffic on an interface.

   outbound-interface *interface-type interface-name*

   By default, a NAT rule is not applied to the outgoing traffic on an interface.

6. Specify an object group for the NAT rule. Choose the options to configure as needed:
   - Specify a source IP address object group.

     source-ip *object-group-name*

     The source IP address object groups is used to match the source IP address in the packet.
   - Specify a destination IP address object group.

     destination-ip *object-group-name*

     The destination IP address object groups is used to match the destination IP address in the packet.
   - Specify a service object group.

     service *object-group-name*

     The service object groups is used to match the service type in the packet.

   By default, the NAT rule does not have any object groups.

   Make sure the specified object groups already exist.

7. Specify an address translation method for the NAT rule. Choose the options to configure as needed:
   - Specify the Easy IP method.

     action easy-ip
   - Specify the NO-NAT method.

     action no-nat
   - Specify the NO-PAT method.

     action address-group { *group-id* | **name** *group-name* } **no-pat** [**reversible**]
   - Specify the PAT method.

action address-group { *group-id* | **name** *group-name* } [**port-preserved**]

By default, no address translation method is specified in a NAT rule.

8. (Optional.) Enable hit counting for the NAT rule.

   counting enable

   By default, hit counting is disabled for the NAT rule.

9. Specify a translation mode for PAT.
   a. Return to interface-based NAT policy view

      quit
   b. Return to system view.

      quit
   c. Apply the Endpoint-Independent Mapping mode for address translation.

      nat mapping-behavior endpoint-independent [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

      By default, the Address and Port-Dependent Mapping mode applies.

      This command takes effect only on outbound PAT.

Rearranging NAT rules in the policy to adjust their priority

About this task

In the interface-based NAT policy, the priority of NAT rules are determined by the configuration order. A rule configured earlier has a higher priority. You can use the **rule move** command to rearrange the NAT rules to adjust their priority.

Restrictions and guidelines

You can use this feature to rearrange only existing NAT rules to change their priority.

Procedure

1. Enter system view.

   system-view
2. Enter the interface-based NAT policy view.

   nat policy
3. Rearrange NAT rules to change their priority.

   rule move *rule-name1* { **after** | **before** } [*rule-name2*]

Disabling NAT rules

Restrictions and guidelines

This feature does not delete a NAT rule, but makes the rule ineffective. You can use the **display nat policy** command to view the status of the NAT rules. If you want to delete a NAT rule, use the **undo rule name** command.

Procedure

1. Enter system view.

   system-view
2. Enter the interface-based NAT policy view.

   nat policy

3. Enter the view of a NAT rule.

   rule name *rule-name*

4. Disable the NAT rule.

   disable

   By default, NAT rules are enabled.

Configuring NAT hairpin

About this task

NAT hairpin allows internal hosts to access each other or allows internal hosts to access internal servers. The source and destination IP addresses of the packets are translated on the interface connected to the internal network.

Restrictions and guidelines

NAT hairpin works in conjunction with the following address translation methods:

- NAT Server and outbound dynamic NAT.
- NAT Server and outbound static NAT.

To provide service correctly, you must configure NAT hairpin on the same interface module as its collaborative NAT features.

When NAT hairpin works in conjunction with NAT Server, you must configure NAT server mappings in one of the following methods with a protocol type specified:

- Configuring common NAT server mappings
- Configuring load sharing NAT server mappings

To configure the P2P mode, you must configure outbound PAT on the interface connected to the external network and enable the EIM mapping mode.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable NAT hairpin.

   nat hairpin enable

   By default, NAT hairpin is disabled.

Configuring NAT ALG

About this task

NAT ALG translates address or port information in the application layer payload to ensure connection establishment.

For example, an FTP application includes a data connection and a control connection. The IP address and port number for the data connection depend on the payload information of the control connection. This requires NAT ALG to translate the address and port information to establish the data connection.

Procedure

1. Enter system view

system-view

2. Configure NAT ALG for a protocol or all protocols.

   nat alg { **all** | **dns** | **ftp** | **h323** | **icmp-error** | **ils** | **mgcp** | **nbt** | **pptp** | **rsh** | **rtsp** | **sccp** | **sip** | **sqlnet** | **tftp** | **xdmcp** }

   By default, NAT ALG is enabled for DNS, FTP, ICMP error messages, PPTP, and RTSP, and is disabled for the other supported protocols.

Configuring NAT DNS mapping

About this task

NAT DNS mapping is applicable to scenarios that internal users accesses an internal server by using the domain name of the server when the DNS server is located at the external network. In the scenario, NAT DNS mapping cooperates with DNS ALG and NAT server mappings.

1. The DNS reply from the external DNS server contains only the domain name and public IP address of the internal server in the payload.
2. The NAT device might have multiple NAT server mappings with the same public IP address but different private IP addresses. DNS ALG might find an incorrect internal server by using only the public IP address. With a NAT DNS mapping is configured, DNS ALG obtains the correct public IP address, public port number, and protocol type of the internal server by using the domain name.
3. A NAT server mapping maps the public IP and port to the private IP and port of the internal server.
4. The NAT device sends the DNS reply to the internal user.

Restrictions and guidelines

NAT DNS mapping works in conjunction with NAT Server. NAT DNS mapping maps the domain name of an internal server to the public IP address, public port number, and protocol type of the internal server. NAT Server maps the public IP and port to the private IP and port of the internal server.

Procedure

1. Enter system view.

   system-view

2. Enable NAT ALG for DNS.

   nat alg dns

   By default, NAT ALG is enabled for DNS.

3. Configure a NAT DNS mapping.

   nat dns-map domain *domain-name* **protocol** *pro-type* { **interface** *interface-type interface-number* | **ip** *global-ip* } **port** *global-port*

   You can configure multiple NAT DNS mappings.

Enabling NAT port halving

About this task

After you enable NAT port halving in VRRP load balancing on an IRF fabric, each port block will be equally divided between the two devices. The two devices will use different ports to translate packets from the same IP address, avoiding port assignment conflicts. For more information about IRF, see *Virtual Technologies Configuration Guide*.

Restrictions and guidelines

Do not use this feature in VRRP standard mode.

Procedure

1. Enter system view.
   `system-view`
2. Enable NAT port halving.
   `nat port-load-balance enable slot` *`slot-number`*
   By default, NAT port halving is disabled.

Configuring hot backup for NAT

About hot backup for NAT

If only one NAT device is deployed in the internal network, internal users cannot access the external network when the NAT device fails. To avoid this situation, configure hot backup for NAT. In the hot backup plan, the two devices in the hot backup system are capable of processing NAT services. Session entries, session relation entries, NAT port block entries, and NAT configurations are synchronized through the hot backup channel. When one device fails, the other device takes over.

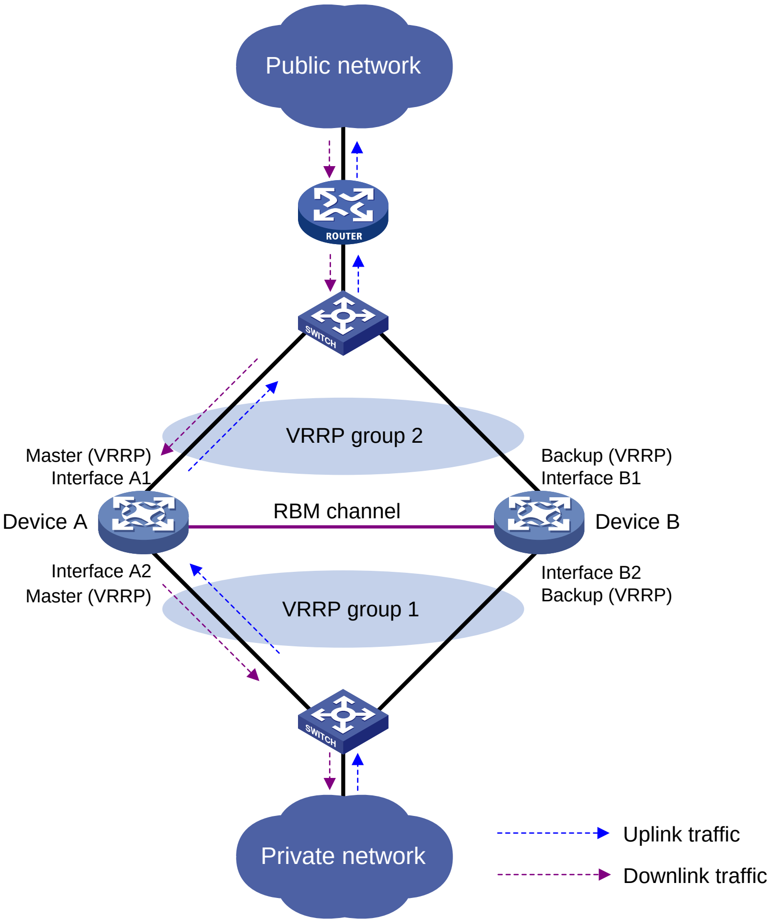
For more information about configuring hot backup, see *High Availability Configuration Guide*.

Operating mechanism

Typically, the master device in the VRRP group processes NAT services in the hot backup system. The following example illustrates how hot backup in active/standby mode ensures uninterrupted NAT services when the master device fails.

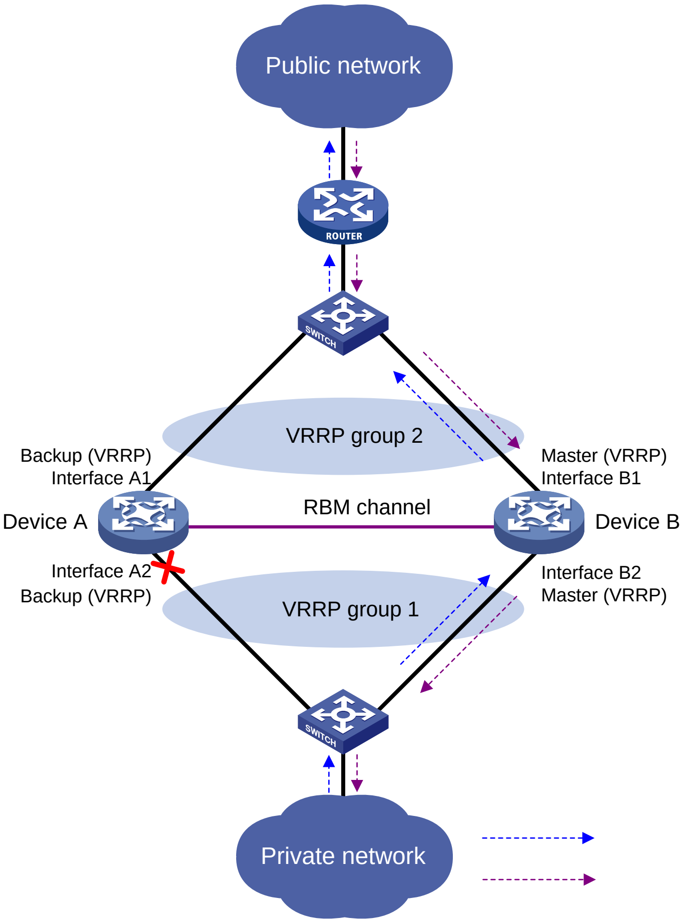
As shown in Figure 17, Device A acts as the primary device and Device B acts as the secondary device in a VRRP group. Device A synchronizes its session entries, session relation entries, and port block entries to Device B in real time through the hot backup channel. Downlinks of Device A and Device B are in VRRP group 1 and uplinks of Device A and Device B are in VRRP group 2. VRRP groups are associated with the hot backup system. Hot backup selects Device A as the master device for address translation based on the link status or forwarding capability of Device A.

Figure 17 Hot backup in active/standby mode



As shown in Figure 18, when Interface A2 of Device A fails, Device B becomes the master device in the VRRP group. Because Device B has NAT configuration information and service entries, NAT services are not interrupted after link switchover.

Figure 18 Traffic switchover in active/standby mode



Configuring active/standby hot backup

About this task

For active/standby hot backup, some translation rules for static address translation, source address translation, and destination address translation assign the public address after translation or public IP address of the internal server to the address management module. Then, both the active and standby devices advertise the mappings between the public IP address and MAC addresses of their own physical interfaces to all nodes in the same LAN. As a result, the uplink Layer 3 device directly connected to the hot backup system might send downlink packets to the backup device, causing service anomalies.

To avoid such an issue, bind an address translation method to a VRRP group. Then, only the master device responds to ARP requests or NS messages with the virtual MAC address of the VRRP group. The uplink Layer 3 device directly connected to the hot backup system sends downlink packets only to the master device.

For more information about configuring the hot backup system, see *High Availability Configuration Guide*.

Restrictions and guidelines

Bind an address translation method to a VRRP group on the primary device in the hot backup system.

Procedure

1. Enter system view.

   system-view

2. Bind a translation method to a VRRP group. Choose the following steps to configure as needed.

3. Bind a VRRP group to the NAT address group.

   a. Enter NAT address group view.

      nat address-group *group-id* [**name** *group-name*]

   b. Bind a VRRP group to the NAT address group.

      vrrp vrid *virtual-router-id*

      By default, a NAT address group is not bound to any VRRP group.

4. Bind a VRRP group to the NAT port block group.

   a. Enter NAT port block group view.

      nat port-block-group *group-id*

   b. Bind a VRRP group to the NAT port block group.

      vrrp vrid *virtual-router-id*

      By default, a NAT port block group is not bound to any VRRP group.

5. Bind a VRRP group to the one-to-one mapping for outbound static NAT.

   a. Enter interface view.

      interface *interface-type interface-number*

   b. Bind a VRRP group to the one-to-one mapping for outbound static NAT.

      For more information, see "Configuring outbound one-to-one static NAT."

6. Bind a VRRP group to the net-to-net mapping for outbound static NAT.

   a. Enter interface view.

      interface *interface-type interface-number*

   b. Bind a VRRP group to the net-to-net mapping for outbound static NAT.

      For more information, see "Configuring outbound net-to-net static NAT."

7. Bind a VRRP group to the object group-based outbound static NAT mapping.

   a. Enter interface view.

      interface *interface-type interface-number*

   b. Bind a VRRP group to the object group-based outbound static NAT mapping.

      For more information, see "Configuring object group-based outbound static NAT."

8. Bind a VRRP group to the internal server.

   a. Enter interface view.

      interface *interface-type interface-number*

   b. Bind a VRRP group to the internal server.

      For more information, see "Configuring common NAT server mappings", "Configuring load sharing NAT server mappings", "Configuring ACL-based NAT server mappings", and "Configuring object group-based NAT server mappings."

Configuring dual-active hot backup

About this task

For dual-active hot backup, the uplink Layer 3 device directly connected to the hot backup system might send downlink packets to the backup device, causing service anomalies.

To avoid such an issue, bind an address translation method to a VRRP group. Then, only the master device responses to ARP requests or NS messages with the virtual MAC address of the VRRP group. For more information about configuring the hot backup system, see *High Availability Configuration Guide*.

Restrictions and guidelines

Select one of the following configuration methods:

- The two devices can share the same NAT address group or port block group. To prevent different master devices from using the same IP-port mapping for different hosts, specify the PAT translation mode and execute the **nat remote-backup port-alloc** command on the primary device.
- As a best practice to prevent different master devices from using the same IP-port mapping for different hosts, configure the two devices to use different public IP addresses for address translation. For example, if the two devices use different NAT address groups or port block groups, user traffic with different source IP addresses is identified by ACLs in NAT rules. To enable different master devices to translate the forward user traffic, specify different gateway addresses for different internal users. To direct the reverse traffic to different master devices, bind NAT address groups or port block groups to different VRRP groups on the primary device.

Procedure

1. Enter system view.

   system-view

2. Bind a translation method to a VRRP group. Choose the following steps to configure as needed.

3. Bind a VRRP group to the NAT address group.

   a. Enter NAT address group view.

      nat address-group *group-id* [**name** *group-name*]

   b. Bind a VRRP group to the NAT address group.

      vrrp vrid *virtual-router-id*

      By default, a NAT address group is not bound to any VRRP group.

      If you execute this command multiple times, the most recent configuration takes effect.

4. Bind a VRRP group to the NAT port block group.

   a. Enter NAT port block group view.

      nat port-block-group *group-id*

   b. Bind a VRRP group to the NAT port block group.

      vrrp vrid *virtual-router-id*

      By default, a NAT port block group is not bound to any VRRP group.

      If you execute this command multiple times, the most recent configuration takes effect.

5. (Optional.) Specify NAT port block ranges for the two devices in the hot backup system.

   a. Return to system view.

      quit

   b. Specify NAT port ranges for the two devices in the hot backup system.

      nat remote-backup port-alloc { **primary** | **secondary** }

      By default, the two devices in the hot backup system share NAT port resources.

      The following table describes port ranges indicated by the keywords:

Keyword	Port ranges
primary	The first half of the port range.

Keyword	Port ranges
`secondary`	The second half of the port range.

6. Bind a VRRP group to the one-to-one mapping for outbound static NAT.
   a. Enter interface view.
      interface *interface-type interface-number*
   b. Bind a VRRP group to the one-to-one mapping for outbound static NAT.
      For more information, see "Configuring outbound one-to-one static NAT."
7. Bind a VRRP group to the net-to-net mapping for outbound static NAT.
   a. Enter interface view.
      interface *interface-type interface-number*
   b. Bind a VRRP group to the net-to-net mapping for outbound static NAT.
      For more information, see "Configuring outbound net-to-net static NAT."
8. Bind a VRRP group to the object group-based outbound static NAT mapping.
   a. Enter interface view.
      interface *interface-type interface-number*
   b. Bind a VRRP group to the object group-based outbound static NAT mapping.
      For more information, see "Configuring object group-based outbound static NAT."
9. Bind a VRRP group to the internal NAT server.
   a. Enter interface view.
      interface *interface-type interface-number*
   b. Bind a VRRP group to the internal NAT server.
      For more information, see "Configuring common NAT server mappings", "Configuring load sharing NAT server mappings", "Configuring ACL-based NAT server mappings", and "Configuring object group-based NAT server mappings."

Configuring NAT maintenance

Configuring periodic NAT statistics collection

About this task

This feature periodically counts sessions and port block assignment failures for address groups.

Hardware and feature compatibility

Models	Feature compatibility
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280	No
NFNX5-HD6480, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	Yes

Restrictions and guidelines

This feature might cause intensive CPU usage. You can disable the feature when CPU resources are insufficient.

Procedure

1. Enter system view.

   `system-view`

2. Enable periodic NAT statistics collection.

   `nat periodic-statistics enable`

   By default, periodic NAT statistics collection is disabled.

3. Configure the interval for periodic NAT statistics collection.

   `nat periodic-statistics interval` *`interval`*

   By default, the interval for periodic NAT statistics collection is 300 seconds.

   A narrower interval indicates intensive CPU usage. As a best practice, use the default interval value.

Enabling statistics collection for NAT session creation rate

About this task

This feature collects information about NAT session creation rates. To view the statistics, use the **`display nat statistics`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enable statistics collection for NAT session creation rate.

   `nat session create-rate enable`

   By default, statistics collection for NAT session creation rate is disabled.

Specifying a probe method for detecting reachability of NAT address group members

About this task

The NAT address group probing uses an NQA template to detect the reachability of the addresses in the group. For information about NQA, see *Network Management and Monitoring Configuration Guide*.

The device periodically sends probe packets to the specified destination address in the NQA template. The source IP addresses in the probe packets are the IP addresses in the NAT address group.

- If the device receives a response packet for a probe, the probed source IP address can be used for address translation.
- If the device does not receive a response packet for a probe, the probed source IP address will be excluded from address translation temporarily. However, in the next NQA operation period, this excluded IP address is also probed. If a response is received in this round, the IP address can be used for address translation.

Restrictions and guidelines

You can specify multiple NQA templates in one NAT address group view. An IP address in the address group is identified as reachable as long as one probe for this IP address succeeds.

This feature is applicable to NAT address groups used for outbound address translation. The manually configured excluded IP addresses are not probed.

Make sure the NQA template used for NAT address group probing does not have source IP address configured.

Procedure

1. Enter system view.

   `system-view`

2. Enter NAT address group view.

   nat address-group *group-id* [**name** *group-name*]

3. Specify a probe method for the NAT address group.

   probe *template-name*

   By default, no probe method is specified for a NAT address group.

   You can specify a nonexistent probe method. The probing takes effect only after you create and configure the NQA template.

Enabling sending ICMP error messages for NAT failures

About this task

By default, sending ICMP error messages upon NAT failures is disabled on the NAT device. Applications using the ICMP protocol cannot be notified when an event occurs. With this feature enabled, the NAT device sends ICMP error messages upon NAT failures for the applications to locate and troubleshoot the failures.

Restrictions and guidelines

Enable this feature for traceroute because the traceroute function requires ICMP error packets.

Procedure

1. Enter system view.

   `system-view`

2. Enable sending ICMP error messages for NAT failures.

   `nat icmp-error reply`

   By default, no ICMP error messages are sent for NAT failures.

Enabling NAT configuration changes to take effect only on new connections

About this task

By default, NAT configuration changes (such as adding, deleting, editing, or moving NAT rules) might cause traffic on an established connection to match a new NAT rule. As a result, you must create a new connection.

Execute the **at configuration-for-new-connection enable** command if you do not want the NAT configuration change to affect existing connections. After you execute this command on the device, it still performs address translation according to the NAT rules before the configuration change for traffic on existing connections. For traffic on new connections, the device matches the traffic according to the priority of NAT rules after the configuration change and performs address translation based on the matching NAT rules.

Procedure

1. Enter system view.

   `system-view`

2. Enable NAT configuration change to take effect only on new connections.

   ```
   nat configuration-for-new-connection enable
   ```

   By default, NAT configuration change taking effect only on new connections is disabled.

Configuring NAT logging

Configuring NAT session logging

About this task

NAT session logging records NAT session information, including translation information and access information.

A NAT device generates NAT session logs for the following events:

- NAT session establishment.
- NAT session removal. This event occurs when you add a configuration with a higher priority, remove a configuration, change ACLs, when a NAT session ages out, or when you manually delete a NAT session.
- Active NAT session logging.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable NAT logging.

   ```
   nat log enable [ acl { ipv4-acl-number | name ipv4-acl-name } ]
   ```

   By default, NAT logging is disabled.

3. Enable NAT session logging.

   - For NAT session establishment events:

     ```
     nat log flow-begin
     ```

   - For NAT session removal events:

     ```
     nat log flow-end
     ```

   - For active NAT flows:

     ```
     nat log flow-active minutes
     ```

   By default, NAT session logging is disabled.

Configuring NAT444 user logging

About this task

NAT444 user logs are used for user tracing. The NAT444 gateway generates a user log whenever it assigns or withdraws a port block. The log includes the private IP address, public IP address, and port block. You can use the public IP address and port numbers to locate the user's private IP address from the user logs.

A NAT444 gateway generates NAT user logs when one of the following events occurs:

- A port block is assigned.

  For the NAT444 static port block mapping, the NAT444 gateway generates a user log when it translates the first connection from a private IP address.

  For the NAT444 dynamic port block mapping, the NAT444 gateway generates a user log when it assigns or extends a port block for a private IP address.

- A port block is withdrawn.

  For the NAT444 static port block mapping, the NAT444 gateway generates a user log when all connections from a private IP address are disconnected.

  For the NAT444 dynamic port block mapping, the NAT444 gateway generates a user log when all the following conditions are met:

  - All connections from a private IP address are disconnected.
  - The port blocks (including the extended ones) assigned to the private IP address are withdrawn.
  - The corresponding mapping entry is deleted.

Prerequisites

Before configuring NAT444 user logging, you must configure the custom NAT444 log generation and outputting features. For more information, see the information center in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable NAT logging.

   `nat log enable` [**`acl`** { *`ipv4-acl-number`* | **`name`** *`ipv4-acl-name`* }]

   By default, NAT logging is disabled.

   The **`acl`** keyword does not take effect on NAT444 user logging.

3. Enable NAT444 user logging. Choose the options to configure as needed:

   - For port block assignment:

     `nat log port-block-assign`

   - For port block withdrawal:

     `nat log port-block-withdraw`

   By default, NAT444 user logging is disabled.

Configuring NAT alarm logging

About this task

Packets that need to be translated are dropped if the NAT resources are not enough. In NO-PAT, the NAT resources refer to the public IP addresses. In EIM PAT, the NAT resources refer to public IP addresses and ports. In NAT444, the NAT resources refer to public IP addresses, port blocks, or ports in port blocks. NAT alarm logging monitors the usage of NAT resources and outputs logs if the NAT resources are not enough.

For NAT444 dynamic port block mappings, an alarm log is generated upon the port block assignment failure or the failure that port resources cannot meet the user address translation requirement.

Restrictions and guidelines

The **`nat log alarm`** command take effect only after you use the **`nat log enable`** command to enable NAT logging.

Prerequisites

Before configuring NAT alarm logging, you must configure the custom NAT log generation and outputting features. For more information, see the information center in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view
2. Enable NAT logging.

   nat log enable [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   By default, NAT logging is disabled.

   The **acl** keyword does not take effect on NAT alarm logging.
3. Enable NAT alarm logging.

   nat log alarm

   By default, NAT alarm logging is disabled.

   An NAT alarm log is output when NAT resources run out.
4. (Optional.) Set the NAT444 port block usage threshold.

   nat log port-block usage threshold *threshold-value*

   By default, the NAT444 port block usage threshold is 90%.

   The system generates alarm logs if the port block usage exceeds the threshold.

Enabling logging for IP usage of a NAT address group in NO-PAT mode

About this task

The system generates a log if the IP usage of a NAT address group exceeds the threshold.

Restrictions and guidelines

This feature takes effect only after you enable NAT logging by using the **nat log enable** command.

Procedure

1. Enter system view.

   system-view
2. Enable NAT logging.

   nat log enable [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* }]

   By default, NAT logging is disabled.

   The **acl** keyword does not take effect on the logging for IP usage of a NAT address group in NO-PAT mode.
3. Enable logging for the IP usage of a NAT address group in NO-PAT mode and set a threshold.

   nat log no-pat ip-usage [**threshold** *value*]

   By default, logging is disabled for the IP usage of a NAT address group.

Configuring NAT in specific networks

Enabling NAT reply redirection

About this task

In some network scenarios, the inbound dynamic NAT is configured with tunneling, and multiple tunnel interfaces use the same NAT address group. In this case, the device will translate the source

IP addresses of packets from different tunnels into the same NAT address before forwarding them. When the forwarding interface receives the reply packets, the device, by default, will not look up the NAT session table. This will cause the incorrect forwarding of the reply packets. To solve the problem, you can enable the NAT reply redirection feature on the forwarding interface. NAT reply redirection allows the interface to use the NAT session table to translate the destination IP addresses for NAT reply packets and find the correct output interfaces for those NATed reply packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable NAT reply redirection.

   nat redirect reply-route enable

   By default, NAT reply redirection is disabled.

Enabling the deletion of timestamps in TCP SYN and SYN ACK packets

About this task

With this feature configured, the system deletes the timestamps from the TCP SYN and SYN ACK packets after dynamic address translation.

If PAT mode is configured on an interface by using **nat inbound** or **nat outbound**, and the tcp_timestams and tcp_tw_recycle function is configured on the TCP server, TCP connections might not be established. To solve the problem, you can shut down the tcp_tw_recycle function or configure the **nat timestamp delete** command.

Procedure

1. Enter system view.

   system-view

2. Enable the deletion of timestamps in TCP SYN and SYN ACK packets

   nat timestamp delete [**vpn-instance** *vpn-instance-name*]

   By default, the deletion of timestamps in TCP SYN and SYN ACK packets is disabled.

   You can enable this feature for multiple VPN instances by repeating the command with different VPN parameters.

Enabling NAT session recreation after link switchover

About this task

This feature is applicable to a WAN network where two interfaces of the NAT device are configured with outbound dynamic NAT rules using different address groups. When the link of one interface fails, traffic on this link is switched to the link of the other interface and the NAT device operates as follows:

- If the two interfaces are in different security zones, the NAT device deletes old session entries after link switchover. When user traffic later arrives, it triggers the NAT session recreation. This mechanism ensures that internal users can access the external network.
- If the two interfaces are in the same security zone, the NAT device retains old session entries after link switchover. Internal users cannot access the external network because the device

uses old session entries to match the user traffic. To avoid this issue, enable this feature to ensure availability of NAT services.

Procedure

1. Enter system view.

   system-view

2. Enable NAT session recreation after link switchover.

   nat link-switch recreate-session

   By default, NAT session recreation is disabled after link switchover.

Display and maintenance commands for interface-based NAT

IMPORTANT:

Support for the **display nat periodic-statistics** command depends on the device model. For more information, see the command reference.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the NAT ALG status for all supported protocols	**display nat alg**
Display all NAT configuration information.	**display nat all**
Display NAT address group information.	**display nat address-group** [*group-id*]
Display NAT DNS mapping configuration.	**display nat dns-map**
Display information about NAT EIM entries.	**display nat eim** [**slot** *slot-number*]
Display inbound dynamic NAT configuration.	**display nat inbound**
Display NAT logging configuration.	**display nat log**
Display information about NAT NO-PAT entries.	**display nat no-pat** [**slot** *slot-number*]
Display IP usage of NAT address groups in NO-PAT mode.	**display nat no-pat ip-usage** [**address-group** { *group-id* | **name** *group-name* } | **object-group** *object-group-name*] [**slot** *slot-number*]
Display outbound dynamic NAT configuration.	**display nat outbound**
Display the interface-based NAT policy configuration.	**display nat policy**
Display NAT server mappings.	**display nat server**
Display internal server group configuration.	**display nat server-group** [*group-id*]

Task	Command
Display NAT sessions.	**display nat session** [[**responder**] { **source-ip** *source-ip* | **destination-ip** *destination-ip* } * [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**verbose**]
Display static NAT mappings.	**display nat static**
Display NAT statistics.	**display nat statistics** [**summary**] [**slot** *slot-number*]
Display static outbound port block mapping rules for NAT444	**display nat outbound port-block-group**
Display NAT port block group configuration.	**display nat port-block-group** [*group-id*]
Display NAT port block mappings.	**display nat port-block** { **dynamic** [**address-group** { *group-id* | **name** *group-name* }] [**ds-lite-b4**] | **static** [**port-block-group** *group-id*] } [**slot** *slot-number*]
Display the port block usage for address groups	**display nat port-block-usage** [**address-group** *group-id*] [**slot** *slot-number*]
Display NAT address group probe information.	**display nat probe address-group** [*group-id*]
Clear NAT counting statistics.	**reset nat count statistics** { **all** | **global-policy**| **global-policy** | **server** | **static** | **static-port-block** }
Clear NAT sessions.	**reset nat session** [**slot** *slot-number*]

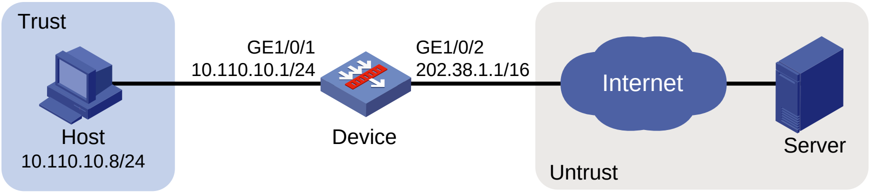
Interface-based NAT configuration examples

Example: Configuring outbound one-to-one static NAT

Network configuration

Configure static NAT to allow the host at 10.110.10.8/24 to access the server at 201.20.1.1/24 on the Internet.

Figure 19 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.2.

   ```
   [Device] ip route-static 201.20.1.0 24 202.38.1.2
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the host to the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-host 10.110.10.8
   [Device-security-policy-ip-1-trust-untrust] destination-ip-host 201.20.1.1
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure NAT:

   # Configure a one-to-one static NAT mapping between the private address 10.110.10.8 and the public address 202.38.1.100.

   ```
   [Device] nat static outbound 10.110.10.8 202.38.1.100
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat static enable
   [Device-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Verify that the host at 10.110.10.8/24 can access the server on the Internet. (Details not shown.)

# Display static NAT configuration.

```
[Device] display nat static
Static NAT mappings:
  Totally 1 outbound static NAT mappings.
  IP-to-IP:
    Local IP     : 10.110.10.8
    Global IP    : 202.38.1.100
    Config status: Active
```

```
Interfaces enabled with static NAT:
  Totally 1 interfaces enabled with static NAT.
  Interface: GigabitEthernet1/0/2
    NAT counting : 0
    Config status: Active
```

# Display NAT sessions.

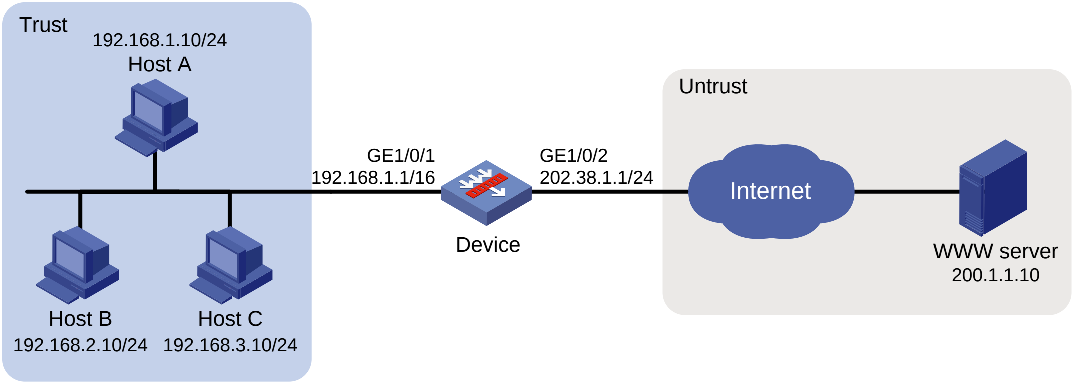
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.110.10.8/54765
  Destination IP/port: 202.38.1.2/23
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.2/23
  Destination IP/port: 202.38.1.100/54765
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: TELNET
Rule ID: -/-/-
Rule name:
Start time: 2017-05-19 10:57:47  TTL: 1195s
Initiator->Responder:            8 packets        375 bytes
Responder->Initiator:           10 packets        851 bytes
Total sessions found: 1
```

Example: Configuring outbound dynamic NAT (non-overlapping addresses)

Network configuration

As shown in Figure 20, a company has a private address 192.168.0.0/16 and two public IP addresses 202.38.1.2 and 202.38.1.3. Configure outbound dynamic NAT to allow only internal users on subnet 192.168.1.0/24 to access the Internet.

Figure 20 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 16
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.20.

   ```
   [Device] ip route-static 200.1.1.0 24 202.38.1.20
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the hosts to the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
   [Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.1.1.10
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure NAT:

   # Configure address group 0, and add an address range from 202.38.1.2 to 202.38.1.3 to the group.

   ```
   [Device] nat address-group 0
   [Device-address-group-0] address 202.38.1.2 202.38.1.3
   ```

```
[Device-address-group-0] quit
```

# Configure ACL 2000 to identify packets from subnet 192.168.1.0/24.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

# Enable outbound dynamic PAT on GigabitEthernet 1/0/2. The source IP addresses of the packets permitted by the ACL rule is translated into the addresses in address group 0.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat outbound 2000 address-group 0
```

Verifying the configuration

# Verify that Host A can access the WWW server, while Host B cannot. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 0
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.3

NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 0
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host A accesses the WWW server.

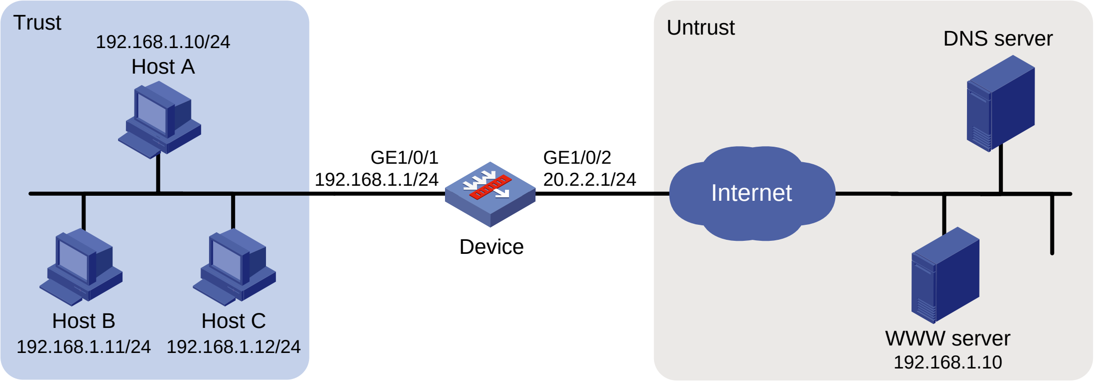
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.10/52082
  Destination IP/port: 200.1.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 200.1.1.10/80
  Destination IP/port: 202.38.1.2/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-19 16:16:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

Example: Configuring outbound bidirectional NAT

Network configuration

As shown in Figure 21, the private network where the Web server resides overlaps with the company private network 192.168.1.0/24. The company has two public IP addresses 202.38.1.2 and 202.38.1.3. Configure NAT to allow internal users to access the external Web server by using the server's domain name.

Figure 21 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure inbound dynamic NAT ALG to make sure the internal host reaches the Web server instead of another internal host. NAT ALG can translate the Web server's IP address in the DNS reply payload to a dynamically assigned public address.
- Configure outbound dynamic NAT to translate the source IP address of packets from an internal host to a dynamically assigned public address.
- Add a static route to the public IP address of the external Web server.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 20.2.2.2.

   ```
   [Device] ip route-static 202.38.1.2 32 20.2.2.2
   ```

4. Configure a security policy:

# Configure a rule named **trust-untrust** to permit the packets from the hosts to the servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 202.38.1.2
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

# Enable NAT ALG for DNS.

```
[Device] nat alg dns
```

# Configure ACL 2000 to identify packets from subnet 192.168.1.0/24.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

# Create address groups.

```
[Device] nat address-group 1
[Device-address-group-1] address 202.38.1.2 202.38.1.2
[Device-address-group-1] quit
[Device] nat address-group 2
[Device-address-group-2] address 202.38.1.3 202.38.1.3
[Device-address-group-2] quit
```

# Enable inbound NO-PAT on GigabitEthernet 1/0/2 to translate the source IP address in the DNS reply payload into the address in address group 1, and allow reversible NAT.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat inbound 2000 address-group 1 no-pat reversible
```

# Enable outbound PAT on GigabitEthernet 1/0/2 to translate the source address of outgoing packets into the address in address group 2.

```
[Device-GigabitEthernet1/0/2] nat outbound 2000 address-group 2
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that Host A can access the Web server by using its domain name. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 2 NAT address groups.
  Address group ID: 1
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.2

  Address group ID: 2
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.3            202.38.1.3

NAT inbound information:
  Totally 1 NAT inbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 1
    Add route: N         NO-PAT: Y         Reversible: Y
    Config status: Active

NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 2
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host A accesses the Web server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.10/51716
  Destination IP/port: 202.38.1.2/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.2/80
  Destination IP/port: 202.38.1.3/1059
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-21 15:36:29  TTL: 1197s
Initiator->Responder:          125 packets       6304 bytes
Responder->Initiator:          223 packets     325718 bytes

Total sessions found: 1
```

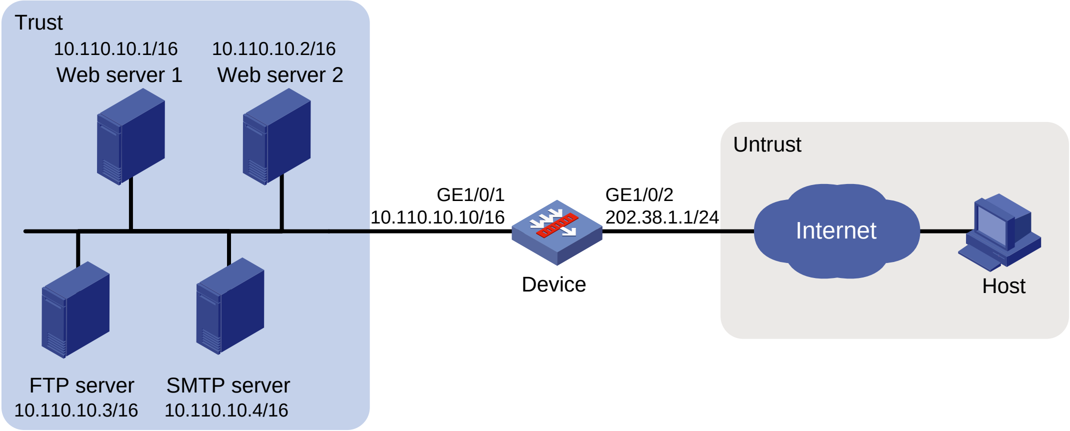
Example: Configuring NAT Server for external-to-internal access

Network configuration

As shown in Figure 22, two Web servers, one FTP server and one SMTP server, are in the internal network to provide services for external users. The internal network address is 10.110.0.0/16. The company has three public IP addresses from 202.38.1.1/24 to 202.38.1.3/24.

Configure the NAT Server feature to allow the external user to access the internal servers with public address 202.38.1.1/24.

Figure 22 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.10 16
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **untrust-trust** to permit the packets from the host to the servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.1
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.2
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.3
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.4
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure NAT:

   # Configure a NAT server mapping to allow external users to access the FTP server by using the address 202.38.1.1 and port 21.

   ```
   [Device] interface gigabitethernet 1/0/2
   ```

```
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.1 21 inside
10.110.10.3 ftp
```

# Configure a NAT server mapping to allow external users to access the Web server 1 by using the address 202.38.1.1 and port 80.

```
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.1 80 inside
10.110.10.1 http
```

# Configure a NAT server mapping to allow external users to access the Web server 2 by using the address 202.38.1.1 and port 8080.

```
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.1 8080 inside
10.110.10.2 http
```

# Configure a NAT server mapping to allow external users to access the SMTP server by using the address 202.38.1.1 and port number defined by SMTP.

```
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.1 smtp inside
10.110.10.4 smtp
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the host on the external network can access the internal servers by using the public addresses. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT internal server information:
  Totally 4 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.1/21
    Local IP/port : 10.110.10.3/21
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active

  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.1/25
    Local IP/port : 10.110.10.4/25
    Rule name     : ServerRule_4
    NAT counting  : 0
    Config status : Active

  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.1/80
    Local IP/port : 10.110.10.1/80
    Rule name     : ServerRule_2
    NAT counting  : 0
    Config status : Active

  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.1/8080
    Local IP/port : 10.110.10.2/80
    Rule name     : ServerRule_3
    NAT counting  : 0
    Config status : Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host accesses the FTP server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 202.38.1.2/52802
  Destination IP/port: 202.38.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.3/21
  Destination IP/port: 202.38.1.2/52802
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-21 11:13:39  TTL: 3597s
Initiator->Responder:            7 packets        313 bytes
Responder->Initiator:            6 packets        330 bytes
Total sessions found: 1
```

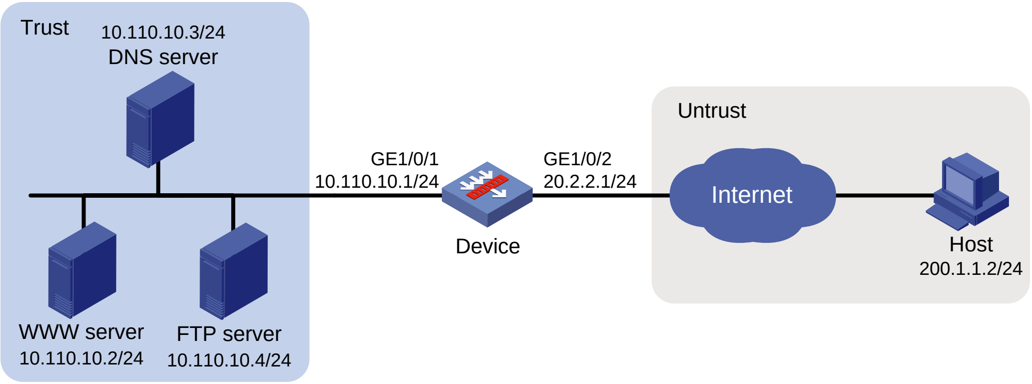
Example: Configuring NAT Server for external-to-internal access through domain name (non-overlapping addresses)

Network configuration

As shown in Figure 23, Web server at 10.110.10.2/24 in the internal network provides services for external users. A DNS server at 10.110.10.3/24 is used to resolve the domain name of the Web server. The company has two public IP addresses: 202.38.1.2 and 202.38.1.3.

Configure NAT Server to allow external users to access the internal Web server by using the domain name.

Figure 23 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure a NAT server mapping to map the private IP address and port of the DNS server to a public address and port. The mapping allows the external host to access the internal DNS server for domain name resolution.
- Enable ALG for DNS and configure outbound dynamic NAT to translate the private IP address of the Web server in the payload of the DNS response packet into a public IP address.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **untrust-trust** to permit the packets from the host to the servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-1-untrust-trust] source-zone untrust
   [Device-security-policy-ip-1-untrust-trust] destination-zone trust
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.2
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.3
   [Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.110.10.4
   [Device-security-policy-ip-1-untrust-trust] action pass
   [Device-security-policy-ip-1-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

4. Configure NAT:

   # Enable NAT with ALG for DNS.

   ```
   [Device] nat alg dns
   ```

   # Configure ACL 2000, and create a rule to permit packets only from 10.110.10.2 to pass through.

   ```
   [Device] acl basic 2000
   [Device-acl-ipv4-basic-2000] rule permit source 10.110.10.2 0
   [Device-acl-ipv4-basic-2000] quit
   ```

   # Create an address group.

   ```
   [Device] nat address-group 1
   [Device-address-group-1] address 202.38.1.3 202.38.1.3
   [Device-address-group-1] quit
   ```

   # Configure a NAT server mapping on GigabitEthernet 1/0/2 to map the address 202.38.1.1 to 10.110.10.3. External users can access the internal DNS server.

   ```
   [Device] interface gigabitethernet 1/0/2
   ```

```
[Device-GigabitEthernet1/0/2] nat server protocol udp global 202.38.1.2 inside
10.110.10.3 dns
```

# Enable outbound NO-PAT on GigabitEthernet 1/0/2. Use the address in address group 1 to translate the private address in DNS response payload, and allow reversible NAT.

```
[Device-GigabitEthernet1/0/2] nat outbound 2000 address-group 1 no-pat reversible
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the host on the external network can access the internal Web server by using the server's domain name. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 1
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.3            202.38.1.3

NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 1
    Port-preserved: N    NO-PAT: Y         Reversible: Y
    Config status: Active

NAT internal server information:
  Totally 1 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 17(UDP)
    Global IP/port: 202.38.1.2/53
    Local IP/port : 10.110.10.3/53
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host accesses Web server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 200.1.1.2/1694
  Destination IP/port: 202.38.1.3/8080
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.2/8080
  Destination IP/port: 200.1.1.2/1694
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: HTTP
```

```
Rule ID: -/-/-
Rule name:
Start time: 2017-06-15 14:53:29  TTL: 3597s
Initiator->Responder:            7 packets        308 bytes
Responder->Initiator:            5 packets        312 bytes
Total sessions found: 1
```

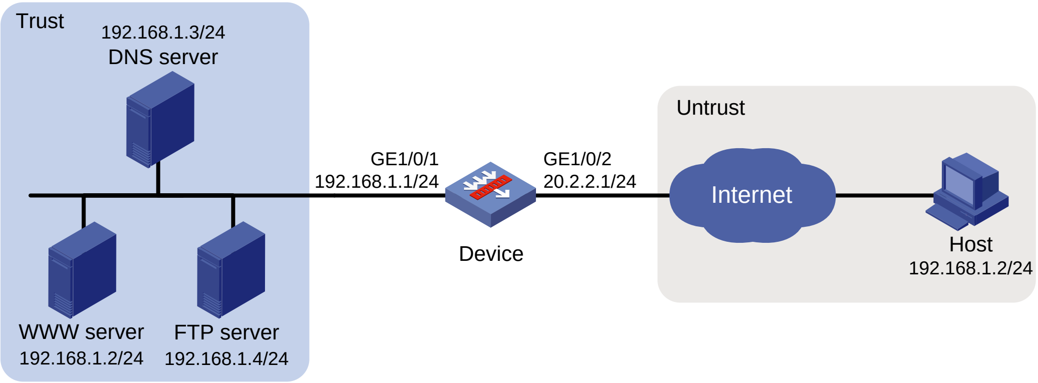
Example: Configuring NAT Server for external-to-internal access through domain name

Network configuration

As shown in Figure 24, an intranet uses the subnet 192.168.1.0/24. The Web server at 192.168.1.2/24 provides Web services for external users and the DNS server at 192.168.1.3/24 resolves the domain name of the Web server. The company has 3 public addresses 202.38.1.2, 202.38.1.3, and 202.38.1.4.

Configure NAT to allow external host at 192.168.1.2 in the external network to use the domain name to access the internal Web server.

Figure 24 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure a NAT server mapping to map the private IP address and port of the DNS server to a public IP address and port. NAT Server allows the external host to access the internal DNS server for domain name resolution.
- Configure outbound dynamic NAT and enable NAT ALG for DNS. The Web server's IP address is the same as the external host's IP address. NAT ALG can translate the Web server's private address in the payload of the DNS response packet to a dynamically assigned public address.
- Configure inbound dynamic NAT. The external host's IP address is the same as the Web server's IP address. Inbound dynamic NAT can translate the external host's IP address into a dynamically assigned public address.
- Add a static route to the public IP address of the external host with GigabitEthernet 1/0/2 as the output interface.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   ```

```
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

This example configures a static route, and the next hop in the routes is 20.2.2.2.

```
[Device] ip route-static 202.38.1.3 32 20.2.2.2
```

4. Configure a security policy:

# Configure a rule named **untrust-trust** to permit the packets from the host to the servers.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.2
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.3
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 192.168.1.4
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

# Enable NAT ALG for DNS.

```
[Device] nat alg dns
```

# Configure ACL 2000 to identify packets from subnet 192.168.1.0/24.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

# Create address groups.

```
[Device] nat address-group 1
[Device-address-group-1] address 202.38.1.2 202.38.1.2
[Device-address-group-1] quit
[Device] nat address-group 2
[Device-address-group-2] address 202.38.1.3 202.38.1.3
[Device-address-group-2] quit
```

# Configure a NAT server mapping on GigabitEthernet 1/0/2 to allow external hosts to access the internal DNS server by using the address 202.38.1.4.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat server protocol udp global 202.38.1.4 inside
192.168.1.3 dns
```

# Enable outbound NO-PAT on GigabitEthernet 1/0/2 to translate IP address of the Web server in the DNS response payload into the address in address group 1, and allow reversible NAT.

```
[Device-GigabitEthernet1/0/2] nat outbound 2000 address-group 1 no-pat reversible
```

# Enable inbound PAT on interface GigabitEthernet 1/0/2 to translate the source address of packets going to the internal network to the address in address group 2.

```
[Device-GigabitEthernet1/0/2] nat inbound 2000 address-group 2
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the host on the external network can use the domain name to access the internal Web server whose address is the same as the host. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 2 NAT address groups.
  Address group ID: 1
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.2

  Address group ID: 2
    Port range: 1-65535
    Address information:
      Start address         End address
      202.38.1.3            202.38.1.3

NAT inbound information:
  Totally 1 NAT inbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 2
    Add route: N         NO-PAT: N         Reversible: N
    Config status: Active

NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 1
    Port-preserved: N    NO-PAT: Y         Reversible: Y
    Config status: Active

NAT internal server information:
  Totally 1 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 17(UDP)
    Global IP/port: 202.38.1.4/53
    Local IP/port : 200.1.1.3/53
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active
```

```
NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host accesses the Web server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.2/1694
  Destination IP/port: 202.38.1.2/8080
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
```

```
    Inbound interface: GigabitEthernet1/0/2
    Source security zone: Untrust
  Responder:
    Source      IP/port: 192.168.1.2/8080
    Destination IP/port: 202.38.1.3/1025
    DS-Lite tunnel peer: -
    VPN instance/VLAN ID/Inline ID: -/-/-
    Protocol: TCP(6)
    Inbound interface: GigabitEthernet1/0/1
    Source security zone: Trust
  State: TCP_ESTABLISHED
  Application: HTTP
  Rule ID: -/-/-
  Rule name:
  Start time: 2017-06-15 14:53:29  TTL: 3597s
  Initiator->Responder:            7 packets        308 bytes
  Responder->Initiator:            5 packets        312 bytes
  Total sessions found: 1
```

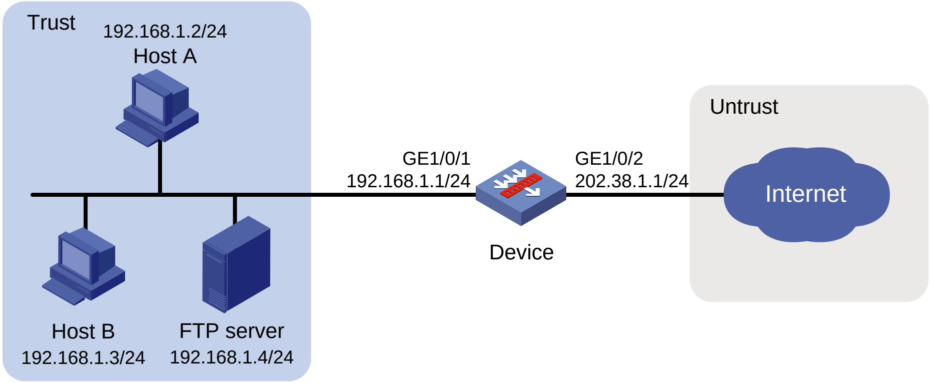
Example: Configuring NAT hairpin in C/S mode

Network configuration

As shown in Figure 25, the internal FTP server at 192.168.1.4/24 provides services for internal and external users. The private network uses two public IP addresses 202.38.1.1 and 202.38.1.2.

Configure NAT hairpin in C/S mode to allow external and internal users to access the internal FTP server by using public IP address 202.38.1.2.

Figure 25 Network diagram



Requirements analysis

To allow external hosts to access the internal FTP server by using a public IP address, configure NAT Server on the interface connected to the external network.

To allow internal hosts to access the internal FTP server by using a public IP address, perform the following tasks:

- Enable NAT hairpin on the interface connected to the internal network.
- Configure outbound NAT on the interface where NAT Server is configured. The destination address is translated by matching the NAT server mapping. The source address is translated by matching the outbound NAT.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **trust-trust** to permit the packets from the hosts to the server in the **Trust** security zone.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-trust
   [Device-security-policy-ip-1-trust-trust] source-zone trust
   [Device-security-policy-ip-1-trust-trust] destination-zone trust
   [Device-security-policy-ip-1-trust-trust] source-ip-host 202.38.1.1
   [Device-security-policy-ip-1-trust-trust] destination-ip-host 192.168.1.4
   [Device-security-policy-ip-1-trust-trust] action pass
   [Device-security-policy-ip-1-trust-trust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **untrust-trust** to permit the packets from the hosts in the **Untrust** security zone to the server.

   ```
   [Device-security-policy-ip] rule name untrust-trust
   [Device-security-policy-ip-2-untrust-trust] source-zone untrust
   [Device-security-policy-ip-2-untrust-trust] destination-zone trust
   [Device-security-policy-ip-2-untrust-trust] destination-ip-host 192.168.1.4
   [Device-security-policy-ip-2-untrust-trust] action pass
   [Device-security-policy-ip-2-untrust-trust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure NAT:

   # Configure ACL 2000 to identify packets from subnet 192.168.1.0/24 to be translated.

   ```
   [Device] acl basic 2000
   [Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
   [Device-acl-ipv4-basic-2000] quit
   ```

   # Configure a NAT server mapping on GigabitEthernet 1/0/2 to map the IP address of the FTP server to a public address, allowing external users to access the internal FTP server.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.2 inside
   192.168.1.4 ftp
   ```

# Enable outbound NAT with Easy IP on GigabitEthernet 1/0/2 so that NAT translates the source addresses of the packets from internal hosts into the IP address of interface GigabitEthernet 1/0/2.

```
[Device-GigabitEthernet1/0/2] nat outbound 2000
```

# Enable NAT hairpin on GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat hairpin enable
[Device-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Verify that both internal and external hosts can access the internal FTP server through the public address. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: ---
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT internal server information:
  Totally 1 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.2/21
    Local IP/port : 192.168.1.4/21
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT hairpinning:
  Totally 1 interfaces enabled with NAT hairpinning.
  Interface: GigabitEthernet1/0/1
    Config status: Active

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host A accesses the FTP server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.2/1694
  Destination IP/port: 202.38.1.2/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 192.168.1.4/21
  Destination IP/port: 202.38.1.1/1025
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: -/-/-
Rule name:
Start time: 2017-06-15 14:53:29  TTL: 3597s
Initiator->Responder:            7 packets        308 bytes
Responder->Initiator:            5 packets        312 bytes
Total sessions found: 1
```

Example: Configuring NAT hairpin in P2P mode

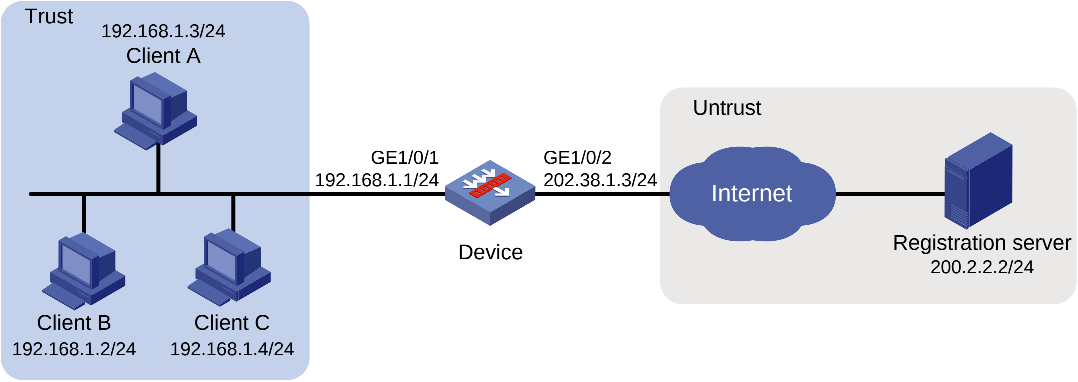
Network configuration

In the P2P application, internal clients must register their IP address to the external server and the server records the registered IP addresses and port numbers of the internal clients. An internal client must request the IP address and port number of another client from the external server before accessing the client.

Configure NAT hairpin so that:

- The internal clients can register the same public address to the external server.
- The internal clients can access each other through the IP address and port number obtained from the server.

Figure 26 Network diagram



Requirements analysis

To meet the network requirements, you must perform the following tasks:

- Configure outbound dynamic PAT on the interface connected to the external network, so the internal clients can access the external server for registration.
- Configure the mapping behavior for PAT as Endpoint-Independent Mapping because the registered IP address and port number should be accessible for any source address.
- Enable NAT hairpin on the interface connected to the internal network so that internal clients can access each other through the public address.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.1.

```
[Device] ip route-static 200.2.2.0 24 202.38.1.1
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the clients to the server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.2.2.2
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

   # Configure a rule named **trust-trust** to permit the packets between the clients in the **Trust** security zone.

```
[Device-security-policy-ip] rule name trust-trust
[Device-security-policy-ip-2-trust-trust] source-zone trust
[Device-security-policy-ip-2-trust-trust] destination-zone trust
[Device-security-policy-ip-2-trust-trust] source-ip-host 202.38.1.3
[Device-security-policy-ip-2-trust-trust] destination-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-2-trust-trust] action pass
[Device-security-policy-ip-2-trust-trust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

   # Configure ACL 2000 to identify packets from subnet 192.168.1.0/24 to be translated.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

   # Configure outbound dynamic PAT with Easy IP on GigabitEthernet 1/0/2. The IP address of GigabitEthernet 1/0/2 is used as the public address for the source address translation of the packets from internal to external.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat outbound 2000
```

   # Configure the Endpoint-Independent Mapping mode for PAT. For packets with the same source address and port number and permitted by ACL 2000, the source address and port number are translated to the same public address and port number.

```
[Device] nat mapping-behavior endpoint-independent acl 2000
```

   # Enable NAT hairpin on GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat hairpin enable
[Device-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Verify that Host A, Host B, and Host C can access each other after they register their IP addresses and port numbers to the external server. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: ---
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT hairpinning:
  Totally 1 interfaces enabled with NAT hairpinning.
  Interface: GigabitEthernet1/0/1
    Config status: Active

NAT mapping behavior:
  Mapping mode : Endpoint-Independent
  ACL          : 2000
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Client A accesses Client B.

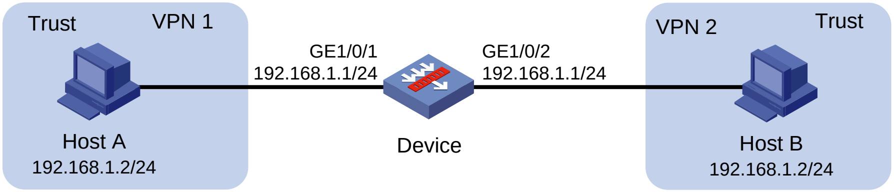
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.3/44929
  Destination IP/port: 202.38.1.3/1
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: UDP(17)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 192.168.1.2/69
  Destination IP/port: 202.38.1.3/1024
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: UDP(17)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: UDP_READY
Application: TFTP
Rule ID: -/-/-
Rule name:
Start time: 2012-08-15 15:53:36  TTL: 46s
Initiator->Responder:            1 packets         56 bytes
Responder->Initiator:            1 packets         72 bytes
Total sessions found: 1
```

Example: Configuring twice NAT

Network configuration

As shown in Figure 27, two departments are in different VPN instances with overlapping addresses. Configure twice NAT so that Host A and Host B in different departments can access each other.

Figure 27 Network diagram



Requirements analysis

Both the source and destination addresses of packets between the two VPNs need to be translated. Configure static NAT on both interfaces connected to the VPNs on the NAT device.

To allow VPNs to access each other, configure the interzone policies to allow VPN packets to pass through.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip binding vpn-instance vpn1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **vpn1-vpn2** to permit the packets from Host A to Host B.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name vpn1-vpn2
   [Device-security-policy-ip-1-vpn1-vpn2] source-zone trust
   [Device-security-policy-ip-1-vpn1-vpn2] destination-zone trust
   [Device-security-policy-ip-1-vpn1-vpn2] source-ip-host 192.168.1.2
   [Device-security-policy-ip-1-vpn1-vpn2] destination-ip-host 192.168.1.2
   [Device-security-policy-ip-1-vpn1-vpn2] vrf vpn1
   [Device-security-policy-ip-1-vpn1-vpn2] action pass
   [Device-security-policy-ip-1-vpn1-vpn2] quit
   ```

   # Configure a rule named **vpn2-vpn1** to permit the packets from Host B to Host A.

   ```
   [Device-security-policy-ip] rule name vpn2-vpn1
   [Device-security-policy-ip-1-vpn2-vpn1] source-zone trust
   [Device-security-policy-ip-1-vpn2-vpn1] destination-zone trust
   [Device-security-policy-ip-1-vpn2-vpn1] source-ip-host 192.168.1.2
   [Device-security-policy-ip-1-vpn2-vpn1] destination-ip-host 192.168.1.2
   [Device-security-policy-ip-1-vpn2-vpn1] vrf vpn2
   [Device-security-policy-ip-1-vpn2-vpn1] action pass
   [Device-security-policy-ip-1-vpn2-vpn1] quit
   ```

```
[Device-security-policy-ip] quit
```

4. Configure NAT:

   # Configure a static outbound NAT mapping between 192.168.1.2 in **vpn 1** and 172.16.1.2 in **vpn 2**.

   ```
   [Device] nat static outbound 192.168.1.2 vpn-instance vpn1 172.16.1.2 vpn-instance vpn2
   ```

   # Configure a static outbound NAT mapping between 192.168.1.2 in **vpn 2** and 172.16.2.2 in **vpn 1**.

   ```
   [Device] nat static outbound 192.168.1.2 vpn-instance vpn2 172.16.2.2 vpn-instance vpn1
   ```

   # Enable static NAT on the two sides of the device.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat static enable
   [Device-GigabitEthernet1/0/2] quit
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] nat static enable
   [Device-GigabitEthernet1/0/1] quit
   ```

Verifying the configuration

# Verify that Host A and Host B can access each other. The public address for Host A is 172.16.1.2 and that for Host B is 172.16.2.2. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
Static NAT mappings:
  Totally 2 outbound static NAT mappings.
  IP-to-IP:
    Local IP     : 192.168.1.2
    Global IP    : 172.16.1.2
    Local VPN    : vpn1
    Global VPN   : vpn2
    Config status: Active

  IP-to-IP:
    Local IP     : 192.168.1.2
    Global IP    : 172.16.2.2
    Local VPN    : vpn2
    Global VPN   : vpn1
    Config status: Active

Interfaces enabled with static NAT:
  Totally 2 interfaces enabled with static NAT.
  Interface: GigabitEthernet1/0/1
    Config status: Active

  Interface: GigabitEthernet1/0/2
    Config status: Active

NAT logging:
  Log enable          : Disabled
```

```
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when Host A accesses Host B.

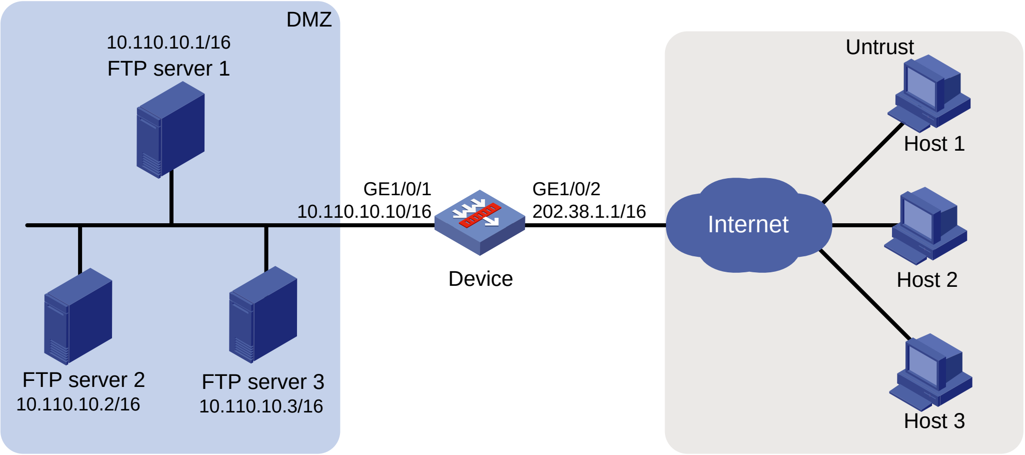
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 192.168.1.2/42496
  Destination IP/port: 172.16.2.2/2048
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: vpn1/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 192.168.1.2/42496
  Destination IP/port: 172.16.1.2/0
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: vpn2/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
State: ICMP_REPLY
Application: INVALID
Rule ID: -/-/-
Rule name:
Start time: 2012-08-16 09:30:49  TTL: 27s
Initiator->Responder:            5 packets        420 bytes
Responder->Initiator:            5 packets        420 bytes
Total sessions found: 1
```

Example: Configuring load sharing NAT Server

Network configuration

As shown in Figure 28, three FTP servers are in the intranet to provide FTP services for external users. Configure NAT so that these external users use the address 202.38.1.1/16 to access the servers and the three FTP servers implement load sharing.

Figure 28 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.110.10.10 16
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/1
   ```

```
[Device-security-zone-DMZ] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure a security policy:

   # Configure a rule named **untrust-dmz** to permit the packets from the hosts to the servers.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name untrust-dmz
   [Device-security-policy-ip-1-untrust-dmz] source-zone untrust
   [Device-security-policy-ip-1-untrust-dmz] destination-zone dmz
   [Device-security-policy-ip-1-untrust-dmz] destination-ip-host 10.110.10.1
   [Device-security-policy-ip-1-untrust-dmz] destination-ip-host 10.110.10.2
   [Device-security-policy-ip-1-untrust-dmz] destination-ip-host 10.110.10.3
   [Device-security-policy-ip-1-untrust-dmz] action pass
   [Device-security-policy-ip-1-untrust-dmz] quit
   [Device-security-policy-ip] quit
   ```

4. Configure NAT:

   # Create NAT server group 0, and add members to the group.

   ```
   [Device] nat server-group 0
   [Device-nat-server-group-0] inside ip 10.110.10.1 port 21
   [Device-nat-server-group-0] inside ip 10.110.10.2 port 21
   [Device-nat-server-group-0] inside ip 10.110.10.3 port 21
   [Device-nat-server-group-0] quit
   ```

   # Associate NAT server group 0 with GigabitEthernet 1/0/2 so that servers in the server group can provide FTP services.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.1 ftp inside
   server-group 0
   [Device-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Verify that external hosts can access the internal FTP server group. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT server group information:
  Totally 1 NAT server groups.
  Group Number      Inside IP             Port    Weight
  0                 10.110.10.1           21      100
                    10.110.10.2           21      100
                    10.110.10.3           21      100

NAT internal server information:
  Totally 1 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.1/21
    Local IP/port : server group 0
                    10.110.10.1/21         (Connections: 1)
```

```
                    10.110.10.2/21         (Connections: 1)
                    10.110.10.3/21         (Connections: 1)
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT sessions that are generated when external hosts access an internal FTP server.

```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 202.38.1.27/5760
  Destination IP/port: 202.38.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.3/21
  Destination IP/port: 202.38.1.27/5760
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: DMZ
State: TCP_ESTABLISHED
Application: FTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-19 16:10:27  TTL: 3598s
Initiator->Responder:           15 packets        702 bytes
Responder->Initiator:           16 packets        891 bytes
Initiator:
  Source      IP/port: 202.38.1.26/30018
  Destination IP/port: 202.38.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.2/21
  Destination IP/port: 202.38.1.26/30018
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: DMZ
State: TCP_ESTABLISHED
Application: FTP
Start time: 2017-05-19 16:09:58  TTL: 3576s
Initiator->Responder:           15 packets        702 bytes
Responder->Initiator:           16 packets        891 bytes
Initiator:
  Source      IP/port: 202.38.1.25/35652
  Destination IP/port: 202.38.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.1/21
  Destination IP/port: 202.38.1.25/35652
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: DMZ
State: TCP_ESTABLISHED
Application: FTP
Start time: 2017-05-19 16:09:46  TTL: 3579s
Initiator->Responder:           15 packets        702 bytes
Responder->Initiator:           16 packets        891 bytes
Total sessions found: 3
```

Example: Configuring NAT DNS mapping

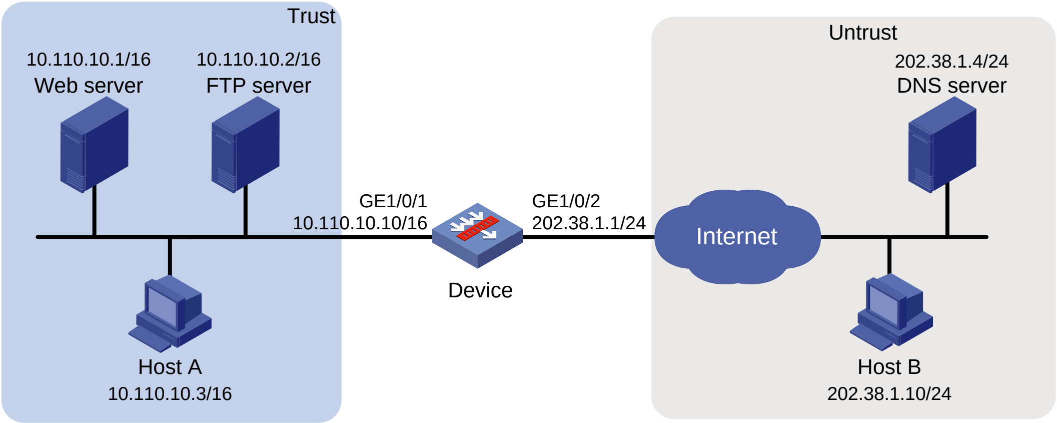
Network configuration

As shown in Figure 29, the internal Web server at 10.110.10.1/16 and FTP server at 10.110.10.2/16 provide services for external user. The company has three public addresses 202.38.1.1 through 202.38.1.3. The DNS server at 202.38.1.4 is on the external network.

Configure NAT so that:

- The public IP address 202.38.1.2 is used by external users to access the Web and FTP servers.
- External users can use the public address or domain name of internal servers to access them.
- Internal users can access the internal servers by using their domain names.

Figure 29 Network diagram



Requirements analysis

To meet the network requirements, perform the following tasks:

- Configure a NAT server mapping by mapping the public IP addresses and port numbers of the internal servers to a public address and port numbers so that external users can access the internal servers.

- Configure NAT DNS mapping and NAT ALG so that the public IP address of the internal server in the payload of the DNS response packet can be translated to the private IP address.

Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.110.10.10 16
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from Host A to the DNS server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 10.110.0.0 16
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 202.38.1.4
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host B to the servers in the **Trust** security zone.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-2-untrust-trust] source-zone untrust
[Device-security-policy-ip-2-untrust-trust] destination-zone trust
[Device-security-policy-ip-2-untrust-trust] destination-ip-host 10.110.10.1
[Device-security-policy-ip-2-untrust-trust] destination-ip-host 10.110.10.2
[Device-security-policy-ip-2-untrust-trust] action pass
[Device-security-policy-ip-2-untrust-trust] quit
```

   # Configure a rule named **trust-trust** to permit the packets from Host A to the servers in the **Trust** security zone.

```
[Device-security-policy-ip] rule name trust-trust
[Device-security-policy-ip-3-trust-trust] source-zone trust
[Device-security-policy-ip-3-trust-trust] destination-zone trust
[Device-security-policy-ip-3-trust-trust] source-ip-host 202.38.1.1
[Device-security-policy-ip-3-trust-trust] destination-ip-host 10.110.10.1
[Device-security-policy-ip-3-trust-trust] destination-ip-host 10.110.10.2
[Device-security-policy-ip-3-trust-trust] action pass
[Device-security-policy-ip-3-trust-trust] quit
[Device-security-policy-ip] quit
```

4. Configure NAT:

# Enable NAT ALG for DNS.

```
[Device] nat alg dns
```

# Configure a NAT server mapping to allow external hosts to access the internal Web server by using the address 202.38.1.2.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.2 inside
10.110.10.1 http
```

# Configure a NAT server mapping to allow external hosts to access the internal FTP server by using the address 202.38.1.2.

```
[Device-GigabitEthernet1/0/2] nat server protocol tcp global 202.38.1.2 inside
10.110.10.2 ftp
```

# Enable outbound NAT with Easy IP on GigabitEthernet 1/0/2.

```
[Device-GigabitEthernet1/0/2] nat outbound
```

# Configure two NAT DNS entries by mapping the domain name **www.server.com** of the Web server to 202.38.1.2, and **ftp.server.com** of the FTP server to 202.38.1.2.

```
[Device] nat dns-map domain www.server.com protocol tcp ip 202.38.1.2 port http
[Device] nat dns-map domain ftp.server.com protocol tcp ip 202.38.1.2 port ftp
[Device] quit
```

Verifying the configuration

# Verify that both internal and external hosts can access the internal servers by using domain names. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: ---
    Address group ID: ---
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT internal server information:
  Totally 2 internal servers.
  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.2/21
    Local IP/port : 10.110.10.2/21
    Rule name     : ServerRule_2
    NAT counting  : 0
    Config status : Active

  Interface: GigabitEthernet1/0/2
    Protocol: 6(TCP)
    Global IP/port: 202.38.1.2/80
    Local IP/port : 10.110.10.1/80
    Rule name     : ServerRule_1
    NAT counting  : 0
    Config status : Active
```

```
NAT DNS mapping information:
  Totally 2 NAT DNS mappings.
  Domain name: ftp.server.com
  Global IP  : 202.38.1.2
  Global port: 21
  Protocol   : TCP(6)
  Config status: Active

  Domain name: www.server.com
  Global IP  : 202.38.1.2
  Global port: 80
  Protocol   : TCP(6)
  Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled
```

```
Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Verify that NAT sessions have been created for external host access to internal Web server.

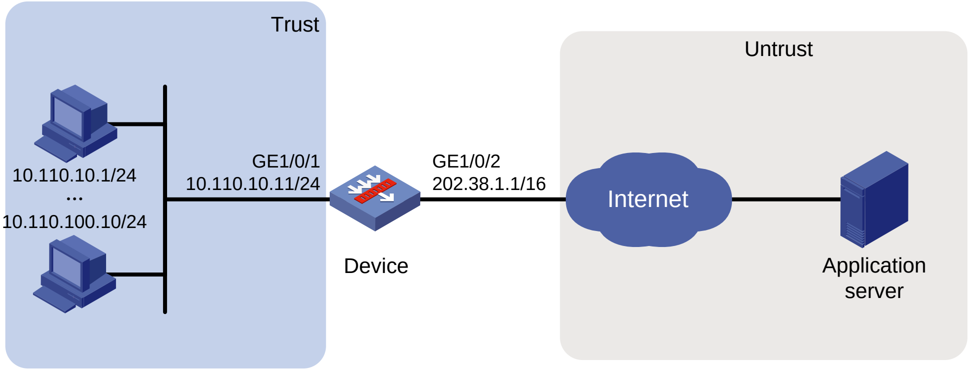
```
[Device] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 202.38.1.10/63593
  Destination IP/port: 202.38.1.2/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: 10.110.10.1/80
  Destination IP/port: 202.38.1.10/63593
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: -/-/-
Rule name:
Start time: 2017-05-21 15:09:11  TTL: 11s
Initiator->Responder:           5 packets       1145 bytes
Responder->Initiator:           3 packets       1664 bytes
Total sessions found: 1
```

Example: Configuring static port block mapping NAT444

Network configuration

As shown in Figure 30, configure static NAT444 to allow users at private IP addresses 10.110.10.1 to 10.110.10.10 to use public IP address 202.38.1.100 for accessing Sever at 200.2.2.1 on the Internet. Configure the port range as 10001 to 15000, and set the port block size to 500.

Figure 30 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.110.10.11 24
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.2.

```
[Device] ip route-static 200.2.2.1 32 202.38.1.2
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the hosts to the application server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-range 10.110.10.1
10.110.10.10
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.2.2.1
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure NAT:

   # Create NAT port block group 1.

```
[Device] nat port-block-group 1
```

   # Add the private IP addresses from 10.110.10.1 to 10.110.10.10 to the port block group.

```
[Device-port-block-group-1] local-ip-address 10.110.10.1 10.110.10.10
```

# Add the public IP address 202.38.1.100 to the port block group.

```
[Device-port-block-group-1] global-ip-pool 202.38.1.100 202.38.1.100
```

# Set the port block size to 500.

```
[Device-port-block-group-1] block-size 500
```

# Configure the port range as 10001 to 15000.

```
[Device-port-block-group-1] port-range 10001 15000
[Device-port-block-group-1] quit
```

# Configure a static outbound port block mapping on GigabitEthernet 1/0/2.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat outbound port-block-group 1
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that users at the private IP addresses can access the Internet. (Details not shown.)

# Display all NAT configuration and statistics.

```
[Device] display nat all
NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

NAT port block group information:
  Totally 1 NAT port block groups.
  Port block group 1:
    Port range: 10001-15000
    Block size: 500
    Local IP address information:
      Start address        End address          VPN instance
      10.110.10.1          10.110.10.10         ---
    Global IP pool information:
      Start address        End address
      202.38.1.100         202.38.1.100

NAT outbound port block group information:
  Totally 1 outbound port block group items.
  Interface: GigabitEthernet1/0/2
    port-block-group: 1
    Config status   : Active

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display static NAT444 mappings.

```
[Device] display nat port-block static
Slot 1:
Local VPN     Local IP         Global IP        Port block   Connections
---           10.110.10.7      202.38.1.100     13001-13500  1
---           10.110.10.5      202.38.1.100     12001-12500  1
---           10.110.10.9      202.38.1.100     14001-14500  1
---           10.110.10.3      202.38.1.100     11001-11500  1
---           10.110.10.2      202.38.1.100     10501-11000  1
---           10.110.10.4      202.38.1.100     11501-12000  1
---           10.110.10.6      202.38.1.100     12501-13000  1
---           10.110.10.1      202.38.1.100     10001-10500  1
---           10.110.10.10     202.38.1.100     14501-15000  1
---           10.110.10.8      202.38.1.100     13501-14000  1
Total mappings found: 10
```

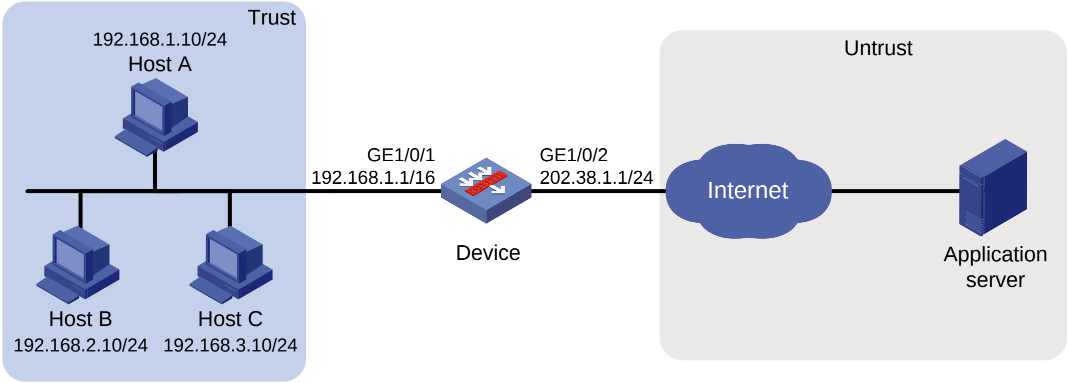
Example: Configuring dynamic port block mapping for NAT444

Network configuration

As shown in Figure 31, a company uses private IP address on network 192.168.0.0/16 and public IP addresses 202.38.1.2 and 202.38.1.3. Configure dynamic NAT444 to meet the following requirements:

- Only users on subnet 192.168.1.0/24 can use public IP addresses 202.38.1.2 and 202.38.1.3 to access the server at 200.2.2.1 on the Internet.
- The port range for the public IP addresses is 1024 to 65535.
- The port block size is 300.
- If the ports in the assigned port block are all used, extend another port block for users.

Figure 31 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 16
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 202.38.1.20.

```
[Device] ip route-static 200.2.2.1 32 202.38.1.20
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from the hosts to the application server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.0 24
[Device-security-policy-ip-1-trust-untrust] destination-ip-host 200.2.2.1
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
```

```
[Device-security-policy-ip] quit
```

5. Configure NAT:

   # Create NAT address group 0.

   ```
   [Device] nat address-group 0
   ```

   # Add public IP addresses 202.38.1.2 and 202.38.1.3 to the NAT address group.

   ```
   [Device-address-group-0] address 202.38.1.2 202.38.1.3
   ```

   # Configure the port range as 1024 to 65535.

   ```
   [Device-address-group-0] port-range 1024 65535
   ```

   # Set the port block size to 300 and the extended port block number to 1.

   ```
   [Device-address-group-0] port-block block-size 300 extended-block-number 1
   [Device-address-group-0] quit
   ```

   # Configure an ACL to identify packets from subnet 192.168.1.0/24.

   ```
   [Device] acl basic 2000
   [Device-acl-ipv4-basic-2000] rule permit source 192.168.1.0 0.0.0.255
   [Device-acl-ipv4-basic-2000] quit
   ```

   # Configure outbound NAT on GigabitEthernet 1/0/2.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat outbound 2000 address-group 0
   [Device-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Verify that Host A can access external servers, but Host B and Host C cannot. (Details not shown.)

# Display all NAT configurations and statistics.

```
[Device] display nat all
NAT address group information:
  Totally 1 NAT address groups.
  Address group ID: 0
    Port range: 1024-65535
    Port block size: 300
    Extended block number: 1
    Address information:
      Start address         End address
      202.38.1.2            202.38.1.3

NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/2
    ACL: 2000
    Address group ID: 0
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active

NAT logging:
  Log enable          : Disabled
  Flow-begin          : Disabled
  Flow-end            : Disabled
  Flow-active         : Disabled
  Port-block-assign   : Disabled
```

```
  Port-block-withdraw : Disabled
  Alarm               : Disabled
  NO-PAT IP usage     : Disabled

NAT mapping behavior:
  Mapping mode : Address and Port-Dependent
  ACL          : ---
  Config status: Active

NAT ALG:
  DNS        : Enabled
  FTP        : Enabled
  H323       : Disabled
  ICMP-ERROR : Enabled
  ILS        : Disabled
  MGCP       : Disabled
  NBT        : Disabled
  PPTP       : Enabled
  RTSP       : Enabled
  RSH        : Disabled
  SCCP       : Disabled
  SCTP       : Disabled
  SIP        : Disabled
  SQLNET     : Disabled
  TFTP       : Disabled
  XDMCP      : Disabled

Static NAT load balancing:     Disabled

NAT link-switch recreate-session: Disabled

NAT configuration-for-new-connection: Disabled
```

# Display NAT statistics.

```
[Device] display nat statistics
  Total session entries: 1
  Session creation rate: 0
  Total EIM entries: 0
  Total inbound NO-PAT entries: 0
  Total outbound NO-PAT entries: 0
  Total static port block entries: 0
  Total dynamic port block entries: 430
  Active static port block entries: 0
  Active dynamic port block entries: 1
```

# Display the dynamic port block entries.

```
[Device] display nat port-block dynamic
Slot 1:
Local VPN     Local IP         Global IP        Port block   Connections
---           192.168.1.10     202.38.1.2       65224-65523  1
```

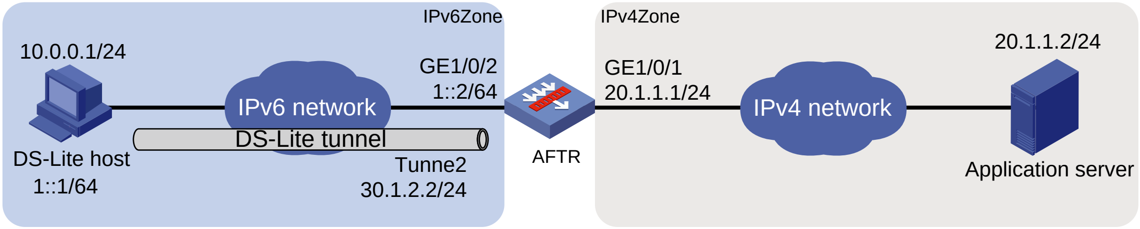
```
Total mappings found: 1
```

Example: Configuring DS-Lite B4 address translation

Network configuration

As shown in Figure 32, configure DS-Lite tunneling and NAT to allow the DS-Lite host to access the IPv4 network over the IPv6 network.

Figure 32 Network diagram



Restrictions and guidelines

Add DS-Lite tunnel interfaces to security zones, and allow traffic between zone pairs to pass through. In this example, Tunnel 2 is added to security zone **IPv6Zone**, and allow traffic between zones **IPv6Zone** and **IPv4Zone**.

Procedure

1. Configure the AFTR:

   a. Assign IP addresses to interfaces:

   # Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 20.1.1.1 24
   [Device-GigabitEthernet1/0/1] quit
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] ipv6 address 1::2 64
   [Device-GigabitEthernet1/0/2] quit
   ```

   # Create a tunnel interface on the AFTR.

   ```
   [Device] interface tunnel 2 mode ds-lite-aftr
   ```

   # Specify an IP address for the tunnel interface.

   ```
   [Device-Tunnel2] ip address 30.1.2.2 255.255.255.0
   ```

   # Specify GigabitEthernet 1/0/2 as the source interface for the tunnel.

   ```
   [Device-Tunnel2] source gigabitethernet 1/0/2
   [Device-Tunnel2] quit
   ```

   b. Add interfaces to security zones.

   ```
   [Device] security-zone name IPv4Zone
   [Device-security-zone-IPv4Zone] import interface gigabitethernet 1/0/1
   [Device-security-zone-IPv4Zone] quit
   [Device] security-zone name IPv6Zone
   [Device-security-zone-IPv6Zone] import interface gigabitethernet 1/0/2
   [Device-security-zone-IPv6Zone] import interface Tunnel 2
   [Device-security-zone-IPv6Zone] quit
   ```

   c. Configure a security policy:

# Configure a rule named **v6-v4** to permit the packets from the host to the application server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name v6-v4
[Device-security-policy-ip-1-v6-v4] source-zone IPv6Zone
[Device-security-policy-ip-1-v6-v4] destination-zone IPv4Zone
[Device-security-policy-ip-1-v6-v4] source-ip-host 10.0.0.1
[Device-security-policy-ip-1-v6-v4] destination-ip-host 20.1.1.2
[Device-security-policy-ip-1-v6-v4] action pass
[Device-security-policy-ip-1-v6-v4] quit
[Device-security-policy-ip] quit
```

# Configure a rule named **v6-local** to allow the device to encapsulate and decapsulate the packets transmitted through the DS-Lite tunnel.

```
[Device] security-policy ipv6
[Device-security-policy-ipv6] rule name v6-local
[Device-security-policy-ipv6-1-v6-local] source-zone IPv6Zone
[Device-security-policy-ipv6-1-v6-local] destination-zone local
[Device-security-policy-ipv6-1-v6-local] source-ip-host 1::1
[Device-security-policy-ipv6-1-v6-local] destination-ip-host 1::2
[Device-security-policy-ipv6-1-v6-local] action pass
[Device-security-policy-ipv6-1-v6-local] quit
[Device-security-policy-ipv6] quit
```

d. Enable DS-Lite tunneling:

# Enable DS-Lite tunneling on GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ds-lite enable
[Device-GigabitEthernet1/0/1] quit
```

e. Configure NAT:

# Create public address group 0.

```
[AFTR] nat address-group 0
```

# Add public IP addresses 20.1.1.11 and 20.1.1.12 to the NAT address group.

```
[AFTR-address-group-0] address 20.1.1.11 20.1.1.12
```

# Configure the port range as 1024 to 65535.

```
[AFTR-address-group-0] port-range 1024 65535
```

# Set the port block size to 300.

```
[AFTR-address-group-0] port-block block-size 300
[AFTR-address-group-0] quit
```

# Configure an IPv6 ACL to identify packets from subnet 1::/64.

```
[AFTR] acl ipv6 basic 2100
[AFTR-acl-ipv4-basic-2100] rule permit source 1::/64
[AFTR-acl-ipv4-basic-2100] quit
```

# Configure DS-Lite NAT444 on GigabitEthernet 1/0/1.

```
[AFTR] interface gigabitethernet 1/0/1
[AFTR-GigabitEthernet1/0/1] nat outbound ds-lite-b4 2100 address-group 0
[AFTR-GigabitEthernet1/0/1] quit
```

2. Configure the DS-Lite host:

# Configure the IPv4 and IPv6 addresses of the DS-Lite host as 10.0.0.1 and 1::1/64. (Details not shown.)

# Configure a static route to the destination IPv4 network. (Details not shown.)

Verifying the configuration

# Use the **display tunnel interface** command to verify that the tunnel interface is up on the AFTR. (Details not shown.)

# Verify that the DS-Lite host can ping the IPv4 application server.

```
C:\> ping 20.1.1.2
Pinging 20.1.1.2 with 32 bytes of data:
Reply from 20.1.1.2: bytes=32 time=51ms TTL=255
Reply from 20.1.1.2: bytes=32 time=44ms TTL=255
Reply from 20.1.1.2: bytes=32 time=1ms TTL=255
Reply from 20.1.1.2: bytes=32 time=1ms TTL=255
Ping statistics for 20.1.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

# Verify that the DS-Lite NAT444 configuration is correct.

```
[AFTR] display nat outbound
NAT outbound information:
  Totally 1 NAT outbound rules.
  Interface: GigabitEthernet1/0/1
    DS-Lite B4 ACL: 2100
    Address group ID: 0
    Port-preserved: N    NO-PAT: N         Reversible: N
    Config status: Active
```

# Verify that the DS-Lite NAT444 configuration takes effect by checking the port block assignment.

```
[AFTR] display nat statistics
Slot 1:
  Total session entries: 1
  Session creation rate: 0
  Total EIM entries: 0
  Total inbound NO-PAT entries: 0
  Total outbound NO-PAT entries: 0
  Total static port block entries: 0
  Total dynamic port block entries: 430
  Active static port block entries: 0
  Active dynamic port block entries: 1
```

# Verify that a NAT444 mapping has been created for the DS-Lite host.

```
[Device] display nat port-block dynamic ds-lite-b4
Slot 1:
Local VPN     DS-Lite B4 addr       Global IP        Port block   Connections
---           1::1                  20.1.1.11        65224-65523  1
Total mappings found: 1
```

Example: Configuring a hot backup system in active/standby mode in collaboration with VRRP for NAT

For more information, see *High Availability Configuration Guide*.

Example: Configuring a hot backup system in dual-active mode in collaboration with VRRP for NAT

For more information, see *High Availability Configuration Guide*.

Contents

Configuring NAT66

Overview

IPv6-to-IPv6 Network Prefix Translation (NPTv6), also known as NAT66, translates the internal IPv6 prefix in the IPv6 packet header to an external IPv6 prefix and vice versa. A device that implements NAT66 translation is called a NAT66 device.

IPv6 source prefix translation

NAT66 source address translation is applicable to the following scenarios:

- **Single internal and external network**—The NAT66 device is connected to an internal network and an external network. Hosts in the internal network uses locally routed IPv6 prefixes. When an internal host sends packets to access the external network, the NAT66 device translates the source IPv6 address prefix in the packets to a global unicast address prefix.
- **Redundancy and load sharing**—Multiple NAT66 devices are deployed between two IPv6 networks and they use ECMPs for load sharing. To allow any NAT66 device to process IPv6 traffic among different sites, configure the same source prefix mappings on these NAT66 devices.
- **Multihoming**—In a multihomed network, NAT66 devices are connected to an internal network and multiple external networks. One internal prefix is mapped to different external prefixes on the NAT66 devices, so that one internal address can be translated to multiple external addresses.

IPv6 destination prefix translation

To allow external users to access internal servers, such as Wed server or FTP server, configure IPv6 destination prefix mappings on the interface connected to the external network.

NAT66 ALG

NAT66 ALG (Application Level Gateway) translates address or port information in the application layer payloads to ensure connection establishment.

For example, an FTP application includes a data connection and a control connection. The IP address and port number for the data connection depend on the payload information of the control connection. This requires NAT66 ALG to translate the address and port information for data connection establishment.

NAT66 ALG supports the following protocol packets: FTP packets and ICMP error messages.

Configuring IPv6 source prefix translation

Restrictions and guidelines

On one interface, the mapping between an internal prefix and an external prefix must be unique.

On different interfaces, different internal prefixes cannot be mapped to the same external prefix.

When you configure source address translation in NO-PAT mode, specify the same IPv6 prefix length before and after NAT66 in a mapping.

The source IPv6 prefix after translation cannot be the same as the external prefix of the NAT66 device or the prefix of the external destination address.

You can configure the global NAT policy to achieve IPv6 source prefix translation. For more information about global NAT policy, see "Configuring NAT."

This feature cannot perform translation on IPsec protected packets with encapsulated by ESP or AH.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure an IPv6 prefix mapping for IPv6 source address translation.

   `nat66 prefix source` *`original-ipv6-prefix prefix-length translated-ipv6-prefix prefix-length`* [**`pat`**]

   By default, no IPv6 prefix mappings are configured for IPv6 source address translation.

Configuring IPv6 destination prefix translation

Restrictions and guidelines

On one interface, the mapping between an external prefix and an internal prefix must be unique.

On different interfaces, one external prefix cannot be mapped to different internal prefixes.

The external IPv6 prefix of the internal server cannot be the same as the external prefix of the NAT66 device or the prefix of external hosts that access the internal server.

You can configure the global NAT policy to achieve IPv6 destination prefix translation. For more information about global NAT policy, see "Configuring NAT."

This feature cannot perform translation on IPsec protected packets with encapsulated by ESP or AH.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure an IPv6 prefix mapping for IPv6 destination address translation.

   `nat66 prefix destination` *`original-ipv6-prefix prefix-length`* [**`protocol`** *`pro-type`* [*`global-port`*]] *`translated-ipv6-prefix prefix-length`* [*`local-port`*]

   By default, no IPv6 prefix mappings are configured for IPv6 destination address translation.

Display and maintenance commands for NAT66

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display all NAT66 configurations.	**`display nat66 all`**
Display NAT66 sessions.	**`display nat66 session`** [**`slot`** *`slot-number`*]

Task	Command
	[**verbose**]
Display NAT66 statistics.	**display nat66 statistics** [**summary**] [**slot** *slot-number*]
Delete NAT66 sessions.	**reset nat66 session** [**slot** *slot-number*]

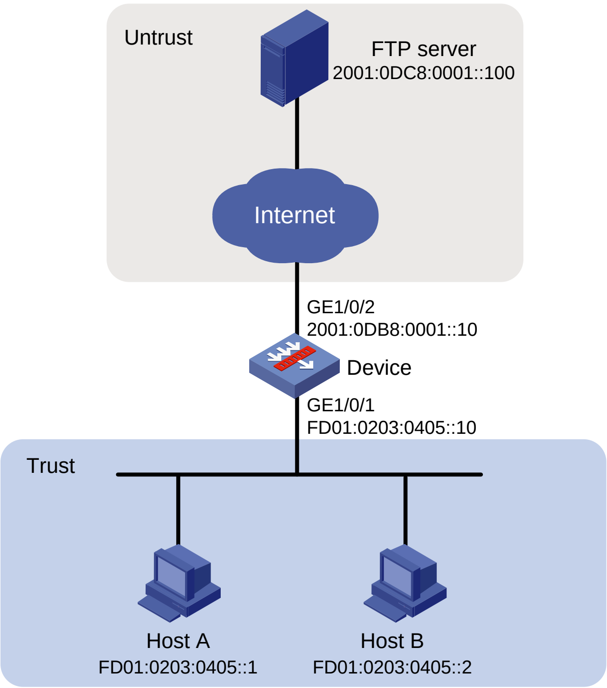
NAT configuration examples

Example: Configuring IPv6 source prefix translation (single internal and external network)

Network configuration

As shown in Figure 1, internal users use IPv6 prefix FD01:0203:0405::/48 and the internal IPv6 addresses are not routable on the Internet. For internal users to access the FTP server on the Internet, configure IPv6 source prefix translation to translate the internal IPv6 prefix FD01:0203:0405::/48 to external prefix 2001:0DF8:0001::/48.

Figure 1 Network diagram



Procedure

1. Assign IPv6 addresses to interfaces:

   # Assign an IPv6 address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ipv6 address FD01:0203:0405::10 48
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 2001:0DB8:0001::11.

   ```
   [Device] ipv6 route-static 2001:0DC8:0001::100 48 2001:0DB8:0001::11
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit the packets from Host A and Host B to the FTP server.

   ```
   [Device] security-policy ipv6
   [Device-security-policy-ipv6] rule 1 name trust-untrust
   [Device-security-policy-ipv6-1-trust-untrust] source-zone trust
   [Device-security-policy-ipv6-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ipv6-1-trust-untrust] source-ip-host FD01:0203:0405::1
   [Device-security-policy-ipv6-1-trust-untrust] source-ip-host FD01:0203:0405::2
   [Device-security-policy-ipv6-1-trust-untrust] destination-ip-host
   2001:0DC8:0001::100
   [Device-security-policy-ipv6-1-trust-untrust] action pass
   [Device-security-policy-ipv6-1-trust-untrust] quit
   [Device-security-policy-ipv6] quit
   ```

5. Configure IPv6 source prefix translation:

   # Configure an IPv6 source prefix mapping from FD01:0203:0405::/48 to 2001:0DF8:0001::/48.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] nat66 prefix source fd01:0203:0405:: 48
   2001:0df8:0001:: 48
   [Device-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Verify that internal hosts can access the FTP server. (Details not shown.)

# Verify NAT66 configurations.

```
[Device] display nat66 all
NAT66 source information:
  Totally 1 source rules.
  Interface(outbound): GigabitEthernet1/0/2
    Original prefix/prefix-length: FD01:203:405::/48
    Translated prefix/prefix-length: 2001:DF8:1::/48
```

# Verify that NAT66 sessions are established.

```
<Device> display nat66 session verbose
Slot 1:
Initiator:
  Source      IP/port: FD01:203:405::1/56002
  Destination IP/port: 2001:DC8:1::100/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
```

```
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 2001:DC8:1::100/21
  Destination IP/port: 2001:DF8:1:D50F::1/56002
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: 1
Rule name: 1
Start time: 2018-12-06 14:48:31  TTL: 3597s
Initiator->Responder:           0 packets          0 bytes
Responder->Initiator:           0 packets          0 bytes

Total sessions found: 1
```

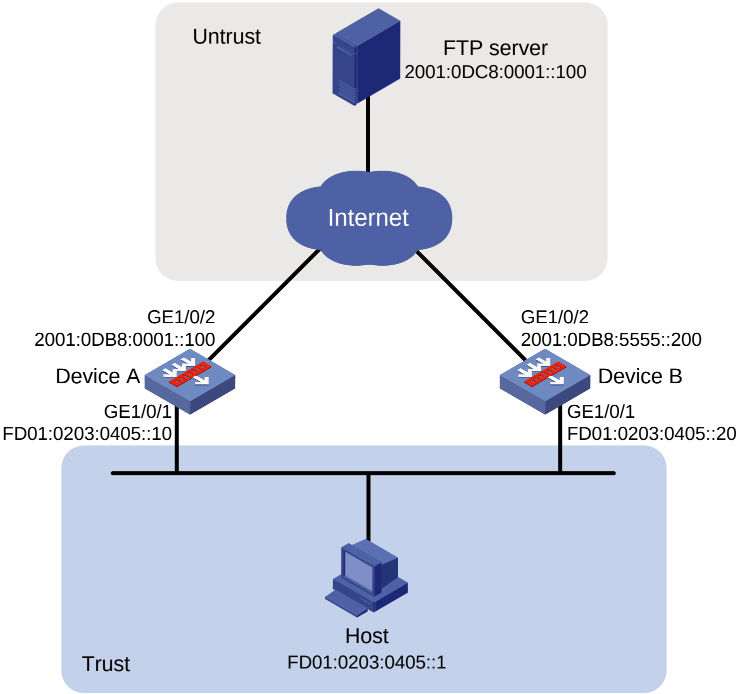
Example: Configuring IPv6 source prefix translation (multihomed network)

Network configuration

As shown in Figure 2, internal users use IPv6 prefix FD01:0203:0405::/48 and the internal IPv6 address is not routable on the Internet.

Device A and Device B are connected to the same internal network but different external networks. Configure IPv6 source prefix translation to translate the internal IPv6 prefix FD01:0203:0405::/48 to different external prefixes so that the internal address can be translated to two external addresses.

Figure 2 Network diagram



Procedure

1. Configure Device A.

a. Assign IPv6 addresses to interfaces:

# Assign an IPv6 address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address FD01:0203:0405::10 48
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

b. Configure settings for routing.

This example configures a static route, and the next hop in the route is 2001:0DB8:0001::11.

```
[DeviceA] ipv6 route-static 2001:0DC8:0001::100 48 2001:0DB8:0001::11
```

c. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
```

d. Configure a security policy:

# Configure a rule named **trust-untrust** to permit the packets from the host to the FTP server.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-zone untrust
```

```
[DeviceA-security-policy-ipv6-1-trust-untrust] source-ip-host FD01:0203:0405::1
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-ip-host 
2001:0DC8:0001::100
[DeviceA-security-policy-ipv6-1-trust-untrust] action pass
[DeviceA-security-policy-ipv6-1-trust-untrust] quit
[DeviceA-security-policy-ipv6] quit
```

e. Configure IPv6 source prefix translation:

# Configure an IPv6 source prefix mapping from FD01:0203:0405::/48 to 2001:0DF8:0001::/48.

```
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] nat66 prefix source fd01:0203:0405:: 48 
2001:0df8:0001:: 48
[DeviceA-GigabitEthernet1/0/2] quit
```

2. Configure Device B.

a. Assign IPv6 addresses to interfaces:

# Assign an IPv6 address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address FD01:0203:0405::20 48
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

b. Configure settings for routing.

This example configures a static route, and the next hop in the route is 2001:0DB8:5555::11.

```
[DeviceB] ipv6 route-static 2001:0DC8:0001::100 48 2001:0DB8:5555::11
```

c. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
```

d. Configure a security policy:

# Configure a rule named **trust-untrust** to permit the packets from the host to the FTP server.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-ip-host FD01:0203:0405::1
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-ip-host 
2001:0DC8:0001::100
[DeviceB-security-policy-ipv6-1-trust-untrust] action pass
[DeviceB-security-policy-ipv6-1-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

e. Configure IPv6 source prefix translation:

# Configure an IPv6 source prefix mapping from FD01:0203:0405::/48 to 2001:0DE8:0001::/48.

```
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] nat66 prefix source fd01:0203:0405:: 48
2001:0de8:0001:: 48
[DeviceB-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that the internal host can access the FTP server through Device A or Device B. The internal prefix is mapped to different external prefixes and external prefixes are mapped to the same internal prefix on the two NAT66 devices.

# Verify NAT66 configurations on Device A.

```
[DeviceA] display nat66 all
NAT66 source information:
  Totally 1 source rules.
  Interface(outbound): GigabitEthernet1/0/2
    Original prefix/prefix-length: FD01:203:405::/48
    Translated prefix/prefix-length: 2001:DF8:1::/48
```

# Verify NAT66 configurations on Device B.

```
[DeviceB] display nat66 all
NAT66 source information:
  Totally 1 source rules.
  Interface(outbound): GigabitEthernet1/0/2
    Original prefix/prefix-length: FD01:203:405::/48
    Translated prefix/prefix-length: 2001:DE8:1::/48
```

# Verify that NAT66 sessions are established on Device A.

```
[DeviceA] display nat66 session verbose
Slot 1:
Initiator:
  Source      IP/port: FD01:203:405::1/35990
  Destination IP/port: 2001:DC8:1::100/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 2001:DC8:1::100/21
  Destination IP/port: 2001:DF8:1:D50F::1/35990
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: 0
Rule name: aaa
Start time: 2021-10-31 14:47:44  TTL: 3584s
Initiator->Responder:            0 packets          0 bytes
Responder->Initiator:            0 packets          0 bytes

Total sessions found: 1
```

# Verify that NAT66 sessions are established on Device B.

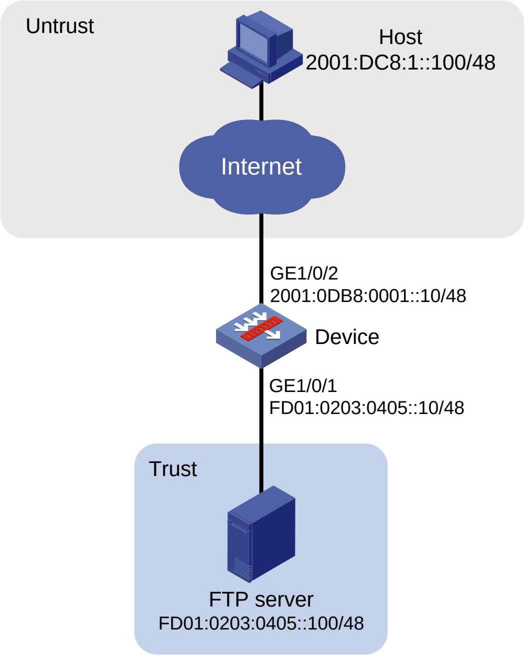
```
[DeviceB] display nat66 session verbose
Slot 1:
Initiator:
  Source      IP/port: FD01:203:405::1/35992
  Destination IP/port: 2001:DC8:1::100/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 2001:DC8:1::100/21
  Destination IP/port: 2001:DE8:1:D51F::1/35992
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: 0
Rule name: aaa
Start time: 2021-10-31 14:50:03  TTL: 3594s
Initiator->Responder:          0 packets          0 bytes
Responder->Initiator:          0 packets          0 bytes

Total sessions found: 1
```

Example: Configuring IPv6 destination prefix translation

Network configuration

As shown in Figure 3, the internal IPv6 address of FTP server is FD01:0203:0405::100/48. The internal prefix is FD01:0203:0405::/48. Configure destination IPv6 prefix translation to allow the FTP server to use IPv6 address 2001:AB01:0001::1 to provide services for external users.

Figure 3 Network diagram



Procedure

1. Assign IPv6 addresses to interfaces:

   # Assign an IPv6 address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ipv6 address FD01:0203:0405::10 48
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **untrust-trust** to permit the packets from the hosts in the external network to the FTP server in the **Trust** security zone.

   ```
   [Device] security-policy ipv6
   [Device-security-policy-ipv6] rule 1 name untrust-trust
   [Device-security-policy-ipv6-1-untrust-trust] source-zone untrust
   [Device-security-policy-ipv6-1-untrust-trust] destination-zone trust
   [Device-security-policy-ipv6-1-untrust-trust] destination-ip-host
   FD01:0203:0405::100
   [Device-security-policy-ipv6-1-untrust-trust] action pass
   [Device-security-policy-ipv6-1-untrust-trust] quit
   [Device-security-policy-ipv6] quit
   ```

4. Configure IPv6 destination prefix translation:

# Configure an IPv6 destination prefix mapping from 2001:AB01:0001::1/128 to FD01:0203:0405::100/128.

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] nat66 prefix destination 2001:ab01:1::1 128
fd01:203:405::100 128
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that external hosts can access the FTP server. (Details not shown.)

# Verify NAT66 configurations.

```
[Device] display nat66 all
NAT66 destination information:
  Totally 1 destination rules.
  Interface(inbound): GigabitEthernet1/0/2
    Original prefix/prefix-length: 2001:AB01:1::1/128
    Translated prefix/prefix-length: FD01:203:405::100/128
```

# Verify that NAT66 sessions are established.

```
[Device] display nat66 session verbose
Slot 1:
Initiator:
  Source      IP/port: 2001:DC8:1::100/9025
  Destination IP/port: 2001:AB01:1::1/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
Responder:
  Source      IP/port: FD01:203:405::100/21
  Destination IP/port: 2001:DC8:1::100/9025
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: 1
Rule name: 1
Start time: 2018-12-06 14:56:03  TTL: 3579s
Initiator->Responder:           0 packets          0 bytes
Responder->Initiator:           0 packets          0 bytes

Total sessions found: 1
```

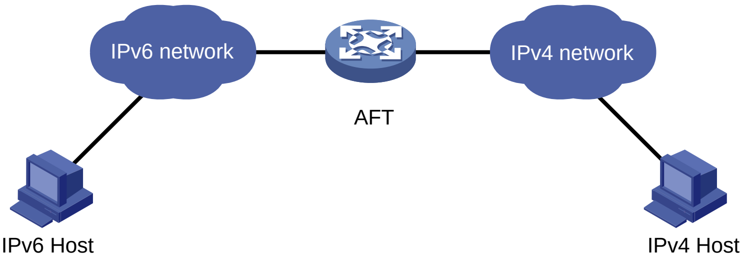
Content

Configuring AFT

About AFT

Address Family Translation (AFT) translates an IP address of one address family into an IP address of the other address family. It enables an IPv4 network and an IPv6 network to communicate with each other, as shown in Figure 1. The IPv4 host and the IPv6 host can communicate with each other without changing the existing configuration.

Figure 1 AFT application scenario



AFT translation methods

Static AFT

Static AFT creates a fixed mapping between an IPv4 address and an IPv6 address.

To perform static AFT, you can configure IPv4-to-IPv6 source address static translation policies and IPv6-to-IPv4 source address static translation policies.

An IPv4-to-IPv6 source address static translation policy can be applied to the following scenarios:

- For an IPv4-initiated session packet, if the source IPv4 address in the packet matches the translation policy, AFT translates the source IPv4 address in the packet to the specified IPv6 address.
- For an IPv6-initiated session packet, if the destination IPv6 address in the packet matches the translation policy, AFT translates the destination IPv6 address in the packet to the specified IPv4 address.

An IPv6-to-IPv4 source address static translation policy can be applied to the following scenarios:

- For an IPv6-initiated session packet, if the source IPv6 address in the packet matches the translation policy, AFT translates the source IPv6 address in the packet to the specified IPv4 address.
- For an IPv4-initiated session packet, if the destination IPv4 address in the packet matches the translation policy, AFT translates the destination IPv4 address in the packet to the specified IPv6 address.

Dynamic AFT

Dynamic AFT creates a dynamic mapping between an IPv4 address and an IPv6 address.

When dynamic AFT performs IPv6-to-IPv4 source address translation, the Not Port Address Translation (NO-PAT) and Port Address Translation (PAT) modes are available.

NO-PAT

NO-PAT translates one IPv6 address to one IPv4 address. An IPv4 address assigned to one IPv6 host cannot be used by any other IPv6 host until it is released.

NO-PAT supports all IP packets.

PAT

PAT translates multiple IPv6 addresses to a single IPv4 address by mapping each IPv6 address and port to the IPv4 address and a unique port. PAT supports the following packet types:

- TCP packets.
- UDP packets.
- ICMPv6 echo request and echo reply messages.

PAT supports port blocks for connection limit and user tracing. Port blocks are generated by dividing the port range (1024 to 65535) by the port block size. Port block based PAT maps multiple IPv6 addresses to one IPv4 address and uses a port block for each IPv6 address.

Port block based PAT functions as follows:

1. When an IPv6 host first initiates a connection to the IPv4 network, it creates a mapping from the host's IPv6 address to an IPv4 address and a port block.
2. It translates the IPv6 address to the IPv4 address, and the source ports to ports in the port block for subsequent connections from the IPv6 host until the ports in the port block are exhausted.

NOTE:

If the port range cannot be divided by the port block size exactly, the remaining ports are not used for translation.

Prefix translation

NAT64 prefix translation

NAT64 prefix is an IPv6 address prefix used to construct an IPv6 address representing an IPv4 node in an IPv6 network. The IPv6 hosts do not use a constructed IPv6 address as their real IP address. The length of a NAT64 prefix can be 32, 40, 48, 56, 64, or 96.

As shown in Figure 2, the construction methods vary depending on the NAT64 prefix length. Bits 64 through 71 in the constructed IPv6 address are reserved bits.

- If the prefix length is 32, 64, or 96 bits, the IPv4 address contained in the IPv6 address will be intact.
- If the prefix length is 40, 48, or 56 bits, the IPv4 address contained in the IPv6 address will be divided into two parts by bits 64 through 71. Bits 64 through 71 are reserved bits and they must be set to 0.

Figure 2 IPv6 address construction with NAT 64 prefix and IPv4 address

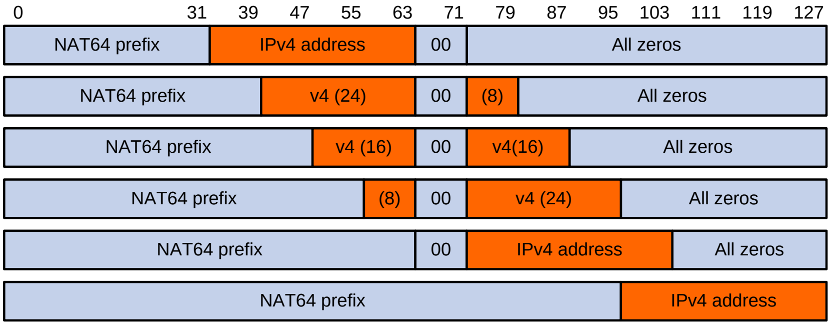


Table 1 shows the examples of addresses constructed by AFT for IPv4 nodes on an IPv6 network.

Table 1 Examples of IPv6 addresses created with an IPv4 address and a NAT64 prefix

IPv6 prefix	IPv4 address	IPv6 address embedded with the IPv4 address
2001:db8::/32	192.0.2.33	2001:db8:c000:221::
2001:db8:100::/40	192.0.2.33	2001:db8:1c0:2:21::
2001:db8:122::/48	192.0.2.33	2001:db8:122:c000:2:2100::
2001:db8:122:300::/56	192.0.2.33	2001:db8:122:3c0:0:221::
2001:db8:122:344::/64	192.0.2.33	2001:db8:122:344:c0:2:2100::
2001:db8:122:344::/96	192.0.2.33	2001:db8:122:344::192.0.2.33

AFT uses a NAT64 prefix to perform the following translation:

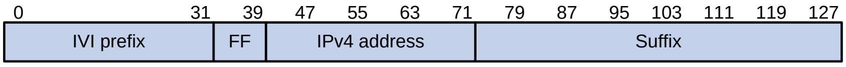
- IPv4-to-IPv6 source address translation. AFT translates a source IPv4 address to an IPv6 address that is created by using the NAT64 prefix and the IPv4 address.
- IPv6-to-IPv4 destination address translation. AFT uses the NAT64 prefix to match destination IPv6 addresses and extracts the embedded IPv4 address from the matching IPv6 addresses.

A NAT64 prefix cannot be on the same subnet as any interface on the device.

IVI prefix translation

An IVI prefix is a 32-bit IPv6 address prefix. An IVI address is the IPv6 address that an IPv6 node uses. As shown in Figure 3, the IVI address includes an IVI prefix and an IPv4 address.

Figure 3 IVI address format



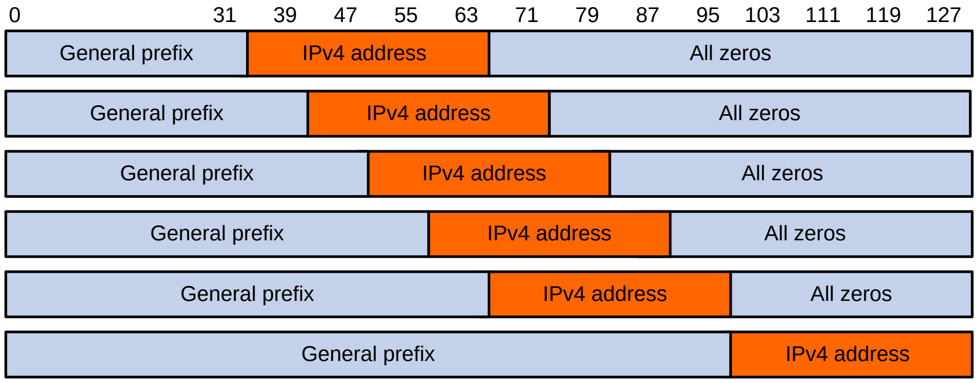
AFT uses an IVI prefix for IPv6-to-IPv4 source address translation. If a source IPv6 address matches the IVI prefix, AFT translates it to the embedded IPv4 address.

General prefix translation

A general prefix is an IPv6 address prefix used to construct an IPv6 address representing an IPv4 node in an IPv6 network. The length of a general prefix can be 32, 40, 48, 56, 64, or 96.

As shown in Figure 4, a general prefix based IPv6 address does not have bits 64 through 71 reserved as a NAT64 prefix based IPv6 address does. An IPv4 address is embedded as a whole into an IPv6 address.

Figure 4 General prefix based IPv6 address format



AFT uses a general prefix for IPv6-to-IPv4 source and destination address translation. If a source or destination IPv6 address matches the general prefix, AFT translates it to the embedded IPv4 address.

A general prefix cannot be on the same subnet as any interface on the device.

IPv6 internal server

IPv6 internal server maps the IPv6 address and port number of an IPv6 internal server to an IPv4 address and port number. It allows the IPv6 internal server to provide services to IPv4 hosts.

IPv4 internal server

IPv4 internal server maps the IPv4 address and port number of an IPv4 internal server to an IPv6 address and port number. It allows the IPv4 internal server to provide services to IPv6 hosts.

AFT translation process

The address translation differs for IPv6-initiated communication and IPv4-initiated communication.

IPv6-initiated communication

As shown in Figure 5, when the IPv6 host initiates access to the IPv4 host, AFT operates as follows:

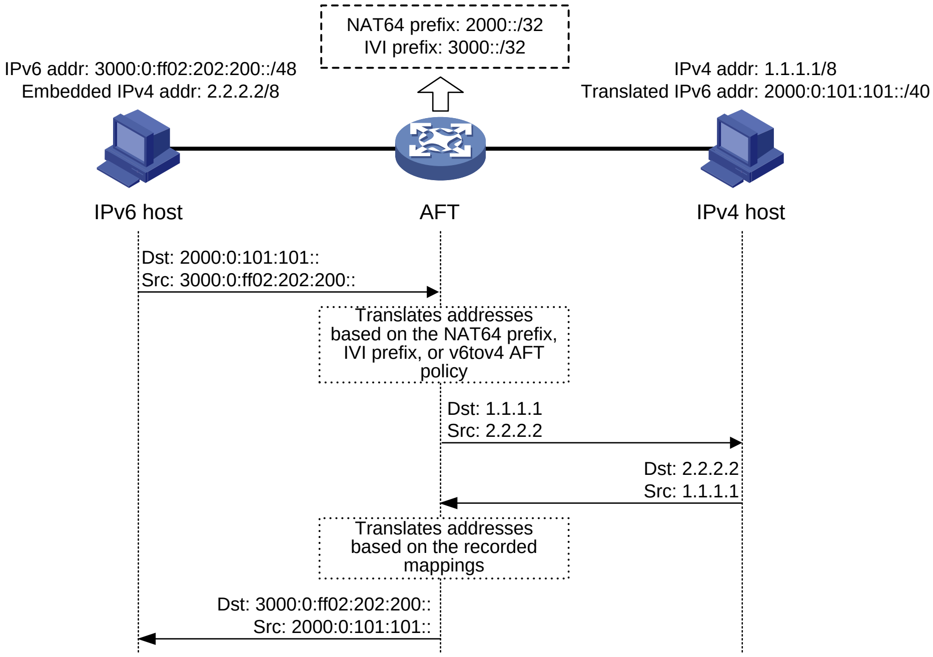
1. Upon receiving a packet from the IPv6 host, AFT compares the packet with IPv6-to-IPv4 destination address translation policies.
    - If a matching policy is found, AFT translates the destination IPv6 address according to the policy.
    - If no matching policy is found, AFT does not process the packet.
2. AFT performs pre-lookup to determine the output interface for the translated packet. PBR is not used for the pre-lookup.
    - If a matching route is found, the process goes to step 3.
    - If no matching route is found, AFT discards the packet.
3. AFT compares the source IPv6 address of the packet with IPv6-to-IPv4 source address translation policies.
    - If a matching policy is found, AFT translates the source IPv6 address according to the policy.
    - If no matching policy is found, AFT discards the packet.

4. AFT forwards the translated packet and records the mappings between IPv6 addresses and IPv4 addresses.
5. AFT translates the IPv4 addresses in the response packet header to IPv6 addresses based on the address mappings before packet forwarding.

For more information about IPv6-to-IPv4 destination address translation policies, see "Configuring an IPv6-to-IPv4 destination address translation policy."

For more information about IPv6-to-IPv4 source address translation policies, see "Configuring an IPv6-to-IPv4 source address translation policy."

Figure 5 AFT process for IPv6-initiated communication


NAT64 prefix: 2000::/32
IVI prefix: 3000::/32
IPv6 addr: 3000:0:ff02:202:200::/48
Embedded IPv4 addr: 2.2.2.2/8
IPv4 addr: 1.1.1.1/8
Translated IPv6 addr: 2000:0:101:101::/40
IPv6 host
AFT
IPv4 host
Dst: 2000:0:101:101::
Src: 3000:0:ff02:202:200::
Translates addresses based on the NAT64 prefix, IVI prefix, or v6tov4 AFT policy
Dst: 1.1.1.1
Src: 2.2.2.2
Dst: 2.2.2.2
Src: 1.1.1.1
Translates addresses based on the recorded mappings
Dst: 3000:0:ff02:202:200::
Src: 2000:0:101:101::

IPv4-initiated communication

As shown in Figure 6, when the IPv4 host initiates access to the IPv6 host, AFT operates as follows:

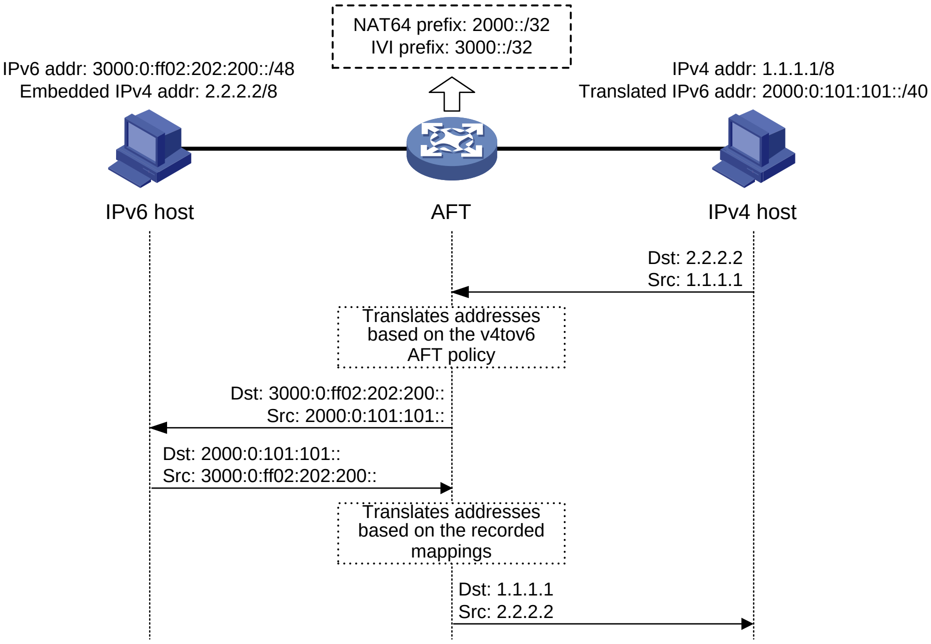
1. Upon receiving a packet from the IPv4 host, AFT compares the packet with IPv4-to-IPv6 destination address translation policies.
   - If a matching policy is found, AFT translates the destination IPv4 address according to the policy.
   - If no matching policy is found, AFT does not perform address translation.
2. AFT performs the pre-lookup to determine output interface for the translated packet. PBR is not used for the pre-lookup.
   - If a matching route is found, the process goes to step 3.
   - If no matching route is found, AFT discards the packet.
3. AFT compares the source IPv4 address with IPv4-to-IPv6 source address translation policies.
   - If a matching policy is found, AFT translates the source IPv4 address according to the policy.
   - If no matching policy is found, AFT discards the packet.
4. AFT forwards the translated packet and records the mappings between IPv4 addresses and IPv6 addresses.

5. AFT translates the IPv6 addresses in the response packet header to IPv4 addresses based on the address mappings before packet forwarding.

For more information about IPv4-to-IPv6 destination address translation policies, see "Configuring an IPv4-to-IPv6 destination address translation policy."

For more information about IPv4-to-IPv6 source address translation policies, see "Configuring an IPv4-to-IPv6 source address translation policy."

Figure 6 AFT process for IPv4-initiated communication


NAT64 prefix: 2000::/32
IVI prefix: 3000::/32
IPv6 addr: 3000:0:ff02:202:200::/48
Embedded IPv4 addr: 2.2.2.2/8
IPv4 addr: 1.1.1.1/8
Translated IPv6 addr: 2000:0:101:101::/40
IPv6 host
AFT
IPv4 host
Dst: 2.2.2.2
Src: 1.1.1.1
Translates addresses based on the v4tov6 AFT policy
Dst: 3000:0:ff02:202:200::
Src: 2000:0:101:101::
Dst: 2000:0:101:101::
Src: 3000:0:ff02:202:200::
Translates addresses based on the recorded mappings
Dst: 1.1.1.1
Src: 2.2.2.2

AFT ALG

AFT ALG translates address or port information in the application layer payloads.

For example, an FTP application includes a data connection and a control connection. The IP address and port number for the data connection depend on the payload information of the control connection. This requires AFT ALG to translate the address and port information.

Restrictions and guidelines: AFT configuration

Packets translated by AFT are not translated by NAT.

AFT tasks at a glance

To configure AFT, perform the following tasks:

1. Enabling AFT
2. Configuring address translation for IPv6-initiated communication
   - Configuring an IPv6-to-IPv4 destination address translation policy
   - Configuring an IPv6-to-IPv4 source address translation policy
   - (Optional.) Setting the ToS field to 0 for translated IPv4 packets
3. Configuring address translation for IPv4-initiated communication

    - Configuring an IPv4-to-IPv6 destination address translation policy
    - Configuring an IPv4-to-IPv6 source address translation policy
    - (Optional.) Setting the Traffic Class field to 0 for translated IPv6 packets
4. (Optional.) Configuring AFT ALG
5. (Optional.) Configuring AFT high availability
    - Enabling AFT port halving for IRF hot backup
    - Associating an AFT address group with a VRRP group
6. (Optional.) Configuring AFT logging

Enabling AFT

Restrictions and guidelines

To implement address translation between IPv4 and IPv6 networks, you must enable AFT on interfaces connected to the IPv4 network and interfaces connected to the IPv6 network.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Enable AFT.
   aft enable
   By default, AFT is disabled.

Configuring an IPv6-to-IPv4 destination address translation policy

About IPv6-to-IPv4 destination address translation policies

AFT compares an IPv6 packet with IPv6-to-IPv4 destination address translation policies in the following order:

1. AFT mappings for IPv4 internal servers.
2. IPv4-to-IPv6 source address static mappings.
3. General prefixes.
4. NAT64 prefixes.

Restrictions and guidelines

Make sure the security policy on the device permits packets that are sent from the IPv6 network-side security zone to security zone **Local**.

Configuring an AFT mapping for an IPv4 internal server

1. Enter system view.
   system-view
2. Configure an AFT mapping for an IPv4 internal server.

aft v4server protocol *protocol-type ipv6-destination-address ipv6-port-number* [**vpn-instance** *ipv6-vpn-instance-name*] *ipv4-destination-address ipv4-port-number* [**vpn-instance** *ipv4-vpn-instance-name*] [**vrrp** *virtual-router-id*]

By default, no AFT mapping for an IPv4 internal server is configured.

Configuring an IPv4-to-IPv6 source address static mapping

Restrictions and guidelines

An IPv4-to-IPv6 source address static translation policy creates a one-to-one mapping between an IPv4 address and an IPv6 address, and can be applied to the following scenarios:

- For an IPv4-initiated session packet, if the source IPv4 address in the packet matches the translation policy, AFT translates the source IPv4 address in the packet to the specified IPv6 address.
- For an IPv6-initiated session packet, if the destination IPv6 address in the packet matches the translation policy, AFT translates the destination IPv6 address in the packet to the specified IPv4 address.

Procedure

1. Enter system view.

   system-view

2. Configure an IPv4-to-IPv6 source address static mapping.

   aft v4tov6 source *ipv4-address* [**vpn-instance** *ipv4-vpn-instance-name*] *ipv6-address* [**vpn-instance** *ipv6-vpn-instance-name*] [**vrrp** *virtual-router-id*]

   By default, no IPv4-to-IPv6 source address static mapping is configured.

Configuring a general prefix

1. Enter system view.

   system-view

2. Configure a general prefix.

   aft prefix-general *prefix-general prefix-length*

   By default, no general prefix is configured.

Configuring a NAT64 prefix

1. Enter system view.

   system-view

2. Configure a NAT64 prefix.

   aft prefix-nat64 *prefix-nat64 prefix-length*

   By default, no NAT64 prefix is configured.

Configuring an IPv6-to-IPv4 source address translation policy

About IPv6-to-IPv4 source address translation policies

AFT compares an IPv6 packet with IPv6-to-IPv4 source address translation policies in the following order:

1. IPv6-to-IPv4 source address static mappings.
2. General prefixes.
3. IVI prefixes.
4. IPv6-to-IPv4 source address dynamic translation policies.

Restrictions and guidelines

Make sure the security policy on the device permits packets that are from security zone **Local** to the IPv4 network-side security zone.

Configuring an IPv6-to-IPv4 source address static mapping

Restrictions and guidelines

An IPv6-to-IPv4 source address static translation policy creates a one-to-one mapping between an IPv6 address and an IPv4 address, and can be applied to the following scenarios:

- For an IPv6-initiated session packet, if the source IPv6 address in the packet matches the translation policy, AFT translates the source IPv6 address in the packet to the specified IPv4 address.
- For an IPv4-initiated session packet, if the destination IPv4 address in the packet matches the translation policy, AFT translates the destination IPv4 address in the packet to the specified IPv6 address.

Procedure

1. Enter system view.

   `system-view`

2. Configure an IPv6-to-IPv4 source address static mapping.

   `aft v6tov4 source` *`ipv6-address`* `[` **`vpn-instance`** *`ipv6-vpn-instance-name`* `]` *`ipv4-address`* `[` **`vpn-instance`** *`ipv4-vpn-instance-name`* `] [` **`vrrp`** *`virtual-router-id`* `]`

Configuring a general prefix

1. Enter system view.

   `system-view`

2. Configure a general prefix.

   `aft prefix-general` *`prefix-general prefix-length`*

Configuring an IVI prefix

1. Enter system view.

   `system-view`

2. Configure an IVI prefix for IPv6-to-IPv4 source address translation.

   aft prefix-ivi *prefix-ivi*

Configuring an IPv6-to-IPv4 source address dynamic translation policy

About this task

An IPv6-to-IPv4 source address dynamic translation policy uses dynamic IPv6-to-IPv4 mappings for IPv6-to-IPv4 source address translation. In PAT mode, AFT translates multiple IPv6 addresses to a single IPv4 address by mapping each IPv6 address and port to the IPv4 address and a unique port. PAT supports the following packet types:

- TCP packets.
- UDP packets.
- ICMPv6 echo request and echo reply messages.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure an AFT address group.

   a. Create an AFT address group and enter AFT address group view.

      aft address-group *group-id*

      This step is required if you decide to use an address group in an IPv6-to-IPv4 source address dynamic translation policy.

   b. Add an address range to the address group.

      address *start-address end-address*

      You can add multiple address ranges to an address group, but the address ranges must not overlap.

   c. Return to system view.

      quit

3. Configure an IPv6-to-IPv4 source address dynamic translation policy.

   aft v6tov4 source { **acl ipv6** { **name** *ipv6-acl-name* | **number** *ipv6-acl-number* } | **prefix-nat64** *prefix-nat64 prefix-length* [**vpn-instance** *ipv6-vpn-instance-name*] } { **address-group** *group-id* [**no-pat** | **port-block-size** *blocksize*] | **interface** *interface-type interface-number* } [**vpn-instance** *ipv4-vpn-instance-name*]

Configuring an IPv4-to-IPv6 destination address translation policy

About IPv4-to-IPv6 destination address translation policies

AFT compares an IPv4 packet with IPv4-to-IPv6 destination address translation policies in the following order:

1. AFT mappings for IPv6 internal servers.
2. IPv6-to-IPv4 source address static mappings.
3. IPv4-to-IPv6 destination address translation policies that use IVI prefixes or general prefixes.

Restrictions and guidelines for configuring an IPv4-to-IPv6 destination address translation policy

Make sure the security policy on the device permits packets that are sent from the IPv4 network-side security zone to security zone **Local**.

Configuring an AFT mapping for an IPv6 internal server

1. Enter system view.

   system-view

2. Configure an AFT mapping for an IPv6 internal server.

   aft v6server protocol *protocol-type ipv4-destination-address ipv4-port-number* [**vpn-instance** *ipv4-vpn-instance-name*] *ipv6-destination-address ipv6-port-number* [**vpn-instance** *ipv6-vpn-instance-name*] [**vrrp** *virtual-router-id*]

Configuring an IPv6-to-IPv4 source address static mapping

Restrictions and guidelines

An IPv6-to-IPv4 source address static translation policy creates a one-to-one mapping between an IPv6 address and an IPv4 address, and can be applied to the following scenarios:

- For an IPv6-initiated session packet, if the source IPv6 address in the packet matches the translation policy, AFT translates the source IPv6 address in the packet to the specified IPv4 address.
- For an IPv4-initiated session packet, if the destination IPv4 address in the packet matches the translation policy, AFT translates the destination IPv4 address in the packet to the specified IPv6 address.

Procedure

1. Enter system view.

   system-view

2. Configure an IPv6-to-IPv4 source address static mapping.

   aft v6tov4 source *ipv6-address* [**vpn-instance** *ipv6-vpn-instance-name*] *ipv4-address* [**vpn-instance** *ipv4-vpn-instance-name*] [**vrrp** *virtual-router-id*]

Configuring an IPv4-to-IPv6 destination address translation policy based on IVI or general prefix

1. Enter system view.

   system-view

2. Configure an IVI prefix or general prefix. Choose one option as needed:

   - Configure an IVI prefix.

     aft prefix-ivi *prefix-ivi*

   - Configure a general prefix.

     aft prefix-general *prefix-general prefix-length*

3. Configure an IPv4-to-IPv6 destination address translation policy that uses an IVI prefix or a general prefix.

   aft v4tov6 destination acl { **name** *ipv4-acl-name* **prefix-ivi** *prefix-ivi* [**vpn-instance** *ipv6-vpn-instance-name*] | **number** *ipv4-acl-number* { **prefix-general** *prefix-general prefix-length* | **prefix-ivi** *prefix-ivi* [**vpn-instance** *ipv6-vpn-instance-name*] } }

   You can use a nonexistent IVI prefix or general prefix in a policy, but the policy takes effect only after you configure the prefix.

Configuring an IPv4-to-IPv6 source address translation policy

About IPv4-to-IPv6 source address translation policies

AFT compares an IPv4 packet with IPv4-to-IPv6 source address translation policies in the following order:

1. IPv4-to-IPv6 source address static mappings.
2. IPv4-to-IPv6 source address translation policies that use NAT64 prefixes or general prefixes.
3. The first NAT64 prefix.

Restrictions and guidelines for configuring an IPv4-to-IPv6 source address translation policy

Make sure the security policy on the device permits packets that are sent from security zone **Local** to the IPv6 network-side security zone.

Configuring an IPv4-to-IPv6 source address static mapping

Restrictions and guidelines

An IPv4-to-IPv6 source address static translation policy creates a one-to-one mapping between an IPv4 address and an IPv6 address, and can be applied to the following scenarios:

- For an IPv4-initiated session packet, if the source IPv4 address in the packet matches the translation policy, AFT translates the source IPv4 address in the packet to the specified IPv6 address.
- For an IPv6-initiated session packet, if the destination IPv6 address in the packet matches the translation policy, AFT translates the destination IPv6 address in the packet to the specified IPv4 address.

Procedure

1. Enter system view.

   system-view

2. Configure an IPv4-to-IPv6 source address static mapping.

   aft v4tov6 source *ipv4-address* [**vpn-instance** *ipv4-vpn-instance-name*] *ipv6-address* [**vpn-instance** *ipv6-vpn-instance-name*] [**vrrp** *virtual-router-id*]

Configuring an IPv4-to-IPv6 source address translation policy based on NAT64 or general prefix

1. Enter system view.

   system-view

2. Configure a NAT64 prefix or general prefix. Choose one option as needed:
   - Configure a NAT64 prefix.

     aft prefix-nat64 *prefix-nat64 prefix-length*

   - Configure a general prefix.

     aft prefix-general *prefix-general prefix-length*

3. Configure an IPv4-to-IPv6 source address translation policy that uses a NAT64 prefix or general prefixes.

   aft v4tov6 source acl { **name** *ipv4-acl-name* **prefix-nat64** *prefix-nat64 prefix-length* [**vpn-instance** *ipv6-vpn-instance-name*] | **number** *ipv4-acl-number* { **prefix-general** *prefix-general prefix-length* | **prefix-nat64** *prefix-nat64 prefix-length* [**vpn-instance** *ipv6-vpn-instance-name*] } }

   You can use a nonexistent NAT64 prefix or general prefix in a policy, but the policy takes effect only after you configure the prefix.

Configuring a NAT64 prefix

1. Enter system view.

   system-view

2. Configure a NAT64 prefix.

   aft prefix-nat64 *prefix-nat64 prefix-length*

Setting the ToS field to 0 for translated IPv4 packets

About this task

You can set the ToS field value for IPv4 packets translated from IPv6 packets:

- If the value is set to 0, the priority of the IPv4 packets is set to the lowest.
- If the value is kept the same as the Traffic Class field value of original IPv6 packets, the priority is not changed.

Procedure

1. Enter system view.

   system-view

2. Set the ToS field to 0 for IPv4 packets translated from IPv6 packets.

   aft turn-off tos

   By default, the ToS field value of translated IPv4 packets is the same as the Traffic Class field value of original IPv6 packets.

Setting the Traffic Class field to 0 for translated IPv6 packets

About this task

You can set the Traffic Class field value for IPv6 packets translated from IPv4 packets:

- If the value is set to 0, the priority of the IPv6 packets is set to the lowest.
- If the value is kept the same as the ToS field value of original IPv4 packets, the priority is not changed.

Procedure

1. Enter system view.
   `system-view`
2. Set the Traffic Class field to 0 for IPv6 packets translated from IPv4 packets.
   `aft turn-off traffic-class`
   By default, the Traffic Class field value of translated IPv6 packets is the same as the ToS field value of original IPv4 packets.

Configuring AFT ALG

Restrictions and guidelines

In an IRF fabric, AFT configured on physical interfaces does not support ALG.

Procedure

1. Enter system view.
   `system-view`
2. Enable AFT ALG for a protocol or all protocols.
   `aft alg` { **`all`** | **`dns`** | **`ftp`** | **`http`** | **`icmp-error`** }
   By default, AFT ALG is enabled for DNS, FTP, ICMP error messages, and HTTP.

Configuring AFT high availability

About AFT high availability

If only one AFT device is deployed in the internal network, internal users cannot access the external network when the AFT device fails. To avoid this situation, configure dual-device hot backup for AFT. The dual-device hot backup supports the IRF hot backup and HA group schemes. In the two schemes, the two IRF/HA group member devices in dual-active or active/standby mode are capable of processing AFT services. Session entries, session relation entries, AFT port block entries, and AFT settings are synchronized through the backup channel. When one device fails, the other device takes over.

For more information about IRF, see *Virtual Technologies Configuration Guide*.

For more information about the HA group, see *High Availability Configuration Guide*.

Enabling AFT port halving for IRF hot backup

Restrictions and guidelines

AFT supports IRF hot backup in active/standby and dual-active mode. The AFT configuration for IRF hot backup depends on the deployment mode.

- In dual-active mode, if the two IRF member devices in an IRF fabric use the same AFT address group, the devices might map different IPv6 addresses and ports to the same IPv4 address and port. To avoid this situation, enable AFT port halving on the devices. After you enable AFT port halving, each port block will be equally divided between the two devices. The two devices will use different ports to translate packets from different IP addresses, avoiding port assignment conflicts.
- In active/standby mode, you do not need to enable AFT port halving on the IRF member devices.

Procedure

1. Enter system view.

   `system-view`

2. Enable AFT port halving.

   `aft port-load-balance enable slot` *`slot-number`*

   By default, AFT port halving is disabled.

Associating an AFT address group with a VRRP group

About this task

In an HA group network collaborated with VRRP, if the virtual IP address of the VRRP group and public addresses in an AFT address group are on the same subnet, bind the AFT address group with the VRRP group. When receiving ARP requests for the MAC addresses corresponding to these public IP addresses, the master device in the VRRP group returns ARP replies with its virtual MAC address.

For more information about configuring the HA group, see *High Availability Configuration Guide*.

Procedure (dual-active mode)

1. Enter system view.

   `system-view`

2. Enter AFT address group view.

   `aft address-group` *`group-id`*

3. Bind the AFT address group to a VRRP group.

   `vrrp vrid` *`virtual-router-id`*

   By default, an AFT address group is not bound to any VRRP group.

4. Return to system view.

   `quit`

5. Specify AFT port ranges for the two devices in the HA group.

   `aft remote-backup port-alloc` `{` **`primary`** `|` **`secondary`** `}`

   By default, the two devices in the HA group share AFT port resources.

   When the two devices in the HA group use the same AFT address group, execute this command on the primary device.

Procedure (active/standby mode)

1. Enter system view.

   `system-view`

2. Enter AFT address group view.

   `aft address-group` *`group-id`*

3. Bind the AFT address group to a VRRP group.

   `vrrp vrid` *`virtual-router-id`*

   By default, an AFT address group is not bound to any VRRP group.

   Execute this command on the primary device in the HA group.

Configuring AFT logging

About this task

For security auditing, you can configure AFT logging to record AFT session information. AFT sessions refer to sessions whose source and destination addresses have been translated by AFT.

AFT can log the following events:

- An AFT port block is created.
- An AFT port block is deleted.
- An AFT session is established.
- An AFT session is removed.

The logs are sent to the information center of the device. For the logs to be output correctly, you must also configure the information center on the device. For more information about information center configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable AFT logging.

   `aft log enable`

   By default, AFT logging is disabled.

   After you configure this command, AFT logs the creation and deletion events of AFT port blocks.

3. (Optional.) Enabling AFT session establishment and removal logging.

   - Enable AFT session establishment logging.

     `aft log flow-begin`

     By default, AFT session establishment logging is disabled.

     AFT session establishment logging takes effect only after you execute the **`aft log enable`** command to enable AFT logging.

   - Enable AFT session removal logging.

     `aft log flow-end`

     By default, AFT session removal logging is disabled.

     AFT session removal logging takes effect only after you execute the **`aft log enable`** command to enable AFT logging.

Display and maintenance commands for AFT

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display AFT configuration.	**display aft configuration**
Display AFT address group information.	**display aft address-group** [*group-id*]
Display AFT mappings.	**display aft address-mapping** [**slot** *slot-number*]
Display information about AFT NO-PAT entries.	**display aft no-pat** [**slot** *slot-number*]
Display AFT port block mappings.	**display aft port-block** [**slot** *slot-number*]
Display information about AFT sessions.	**display aft session ipv4** [{ **source-ip** *source-ip-address* | **destination-ip** *destination-ip-address* } * [**vpn-instance** *ipv4-vpn-instance-name*]] [**slot** *slot-number*] [**verbose**] **display aft session ipv6** [{ **source-ip** *source-ipv6-address* | **destination-ip** *destination-ipv6-address* } * [**vpn-instance** *ipv6-vpn-instance-name*]] [**slot** *slot-number*] [**verbose**]
Display AFT statistics.	**display aft statistics** [**slot** *slot-number*]
Clear AFT sessions.	**reset aft session** [**slot** *slot-number*]
Clear AFT statistics.	**reset aft statistics** [**slot** *slot-number*]

AFT configuration examples

Example: Allowing IPv4 Internet access from an IPv6 network

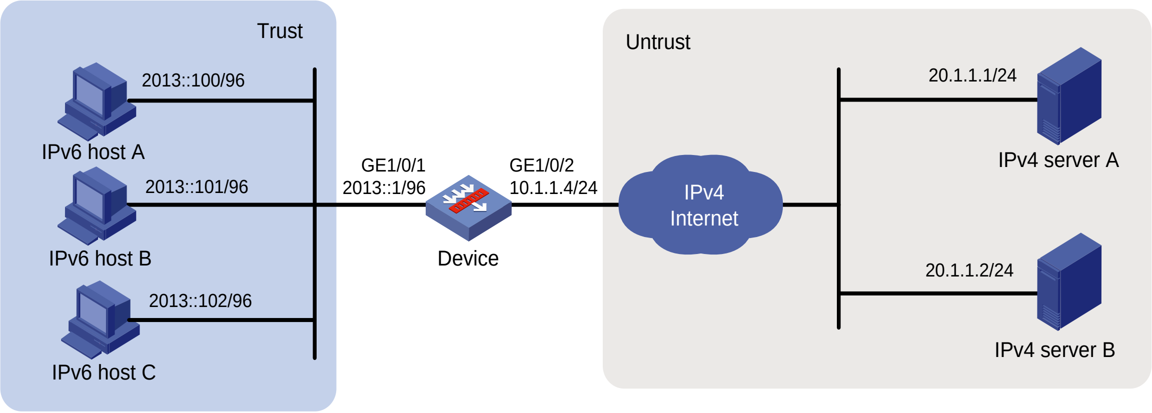
Network configuration

As shown in Figure 7, a company upgrades the network to IPv6 and has IPv4 addresses from 10.1.1.1 to 10.1.1.3.

To allow IPv6 hosts on subnet 2013::/96 to access the IPv4 Internet, configure the following AFT policies on the device:

- Configure a NAT64 prefix to translate IPv4 addresses of IPv4 servers to IPv6 addresses.
- Configure an IPv6-to-IPv4 source address dynamic translation policy to translate source IPv6 addresses of IPv6-initiated packets to IPv4 addresses in the range of 10.1.1.1 to 10.1.1.3.

Figure 7 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ipv6 address 2013::1 96
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

3. Configure settings for routing.

   This example configures a static route with next hop address 10.1.1.100.

   ```
   [Device] ip route-static 20.1.1.0 24 10.1.1.100
   ```

4. Configure security policies:

   # In the IPv6 security policy, configure a rule named **aftlocalin** to allow the device to perform AFT on the IPv6 host traffic destined for the IPv4 servers.

   ```
   [Device] security-policy ipv6
   [Device-security-policy-ipv6] rule name aftlocalin
   [Device-security-policy-ipv6-1-aftlocalin] source-zone trust
   [Device-security-policy-ipv6-1-aftlocalin] destination-zone local
   [Device-security-policy-ipv6-1-aftlocalin] source-ip-subnet 2013:: 96
   [Device-security-policy-ipv6-1-aftlocalin] destination-ip-host 2012::20.1.1.1
   [Device-security-policy-ipv6-1-aftlocalin] destination-ip-host 2012::20.1.1.2
   [Device-security-policy-ipv6-1-aftlocalin] action pass
   [Device-security-policy-ipv6-1-aftlocalin] quit
   [Device-security-policy-ipv6] quit
   ```

   # In the IPv6 security policy, configure a rule named **aftlocalout** to allow the device to forward the AFT-translated packets to the IPv4 servers.

   ```
   [Device] security-policy ip
   ```

```
[Device-security-policy-ip] rule name aftlocalout
[Device-security-policy-ip-1-aftlocalout] source-zone local
[Device-security-policy-ip-1-aftlocalout] destination-zone untrust
[Device-security-policy-ip-1-aftlocalout] source-ip-host 10.1.1.1
[Device-security-policy-ip-1-aftlocalout] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-aftlocalout] source-ip-host 10.1.1.3
[Device-security-policy-ip-1-aftlocalout] destination-ip-host 20.1.1.1
[Device-security-policy-ip-1-aftlocalout] destination-ip-host 20.1.1.2
[Device-security-policy-ip-1-aftlocalout] action pass
[Device-security-policy-ip-1-aftlocalout] quit
[Device-security-policy-ip] quit
```

5. Configure AFT settings:

   # Create AFT address group 0, and add the address range from 10.1.1.1 to 10.1.1.3 to the group.

   ```
   [Device] aft address-group 0
   [Device-aft-address-group-0] address 10.1.1.1 10.1.1.3
   [Device-aft-address-group-0] quit
   ```

   # Configure IPv6 ACL 2000 to permit IPv6 packets only from subnet 2013::/96 to pass through.

   ```
   [Device] acl ipv6 basic 2000
   [Device-acl-ipv6-basic-2000] rule permit source 2013:: 96
   [Device-acl-ipv6-basic-2000] rule deny
   [Device-acl-ipv6-basic-2000] quit
   ```

   # Configure the device to translate source IPv6 addresses of packets permitted by IPv6 ACL 2000 to IPv4 addresses in address group 0.

   ```
   [Device] aft v6tov4 source acl ipv6 number 2000 address-group 0
   ```

   # Configure the device to use NAT64 prefix **2012::/96** to translate destination IPv6 addresses of IPv6 packets.

   ```
   [Device] aft prefix-nat64 2012:: 96
   ```

   # Enable AFT on the interfaces connected to the IPv6 network and IPv4 Internet, respectively.

   ```
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] aft enable
   [Device-GigabitEthernet1/0/1] quit
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] aft enable
   [Device-GigabitEthernet1/0/2] quit
   ```

   # Configure routes to make sure IPv6 hosts can reach IPv6 addresses that are translated by using the NAT64 prefix, and IPv4 servers can reach translated IPv4 addresses. (Details not shown.)

Verifying the configuration

# Verify the connectivity between IPv6 hosts and IPv4 servers. This example pings IPv4 server A from IPv6 host A.

```
D:\>ping 2012::20.1.1.1
Pinging 2012::20.1.1.1 with 32 bytes of data:
Reply from 2012::20.1.1.1: time=3ms
Reply from 2012::20.1.1.1: time=3ms
Reply from 2012::20.1.1.1: time=3ms
Reply from 2012::20.1.1.1: time=3ms
```

# Display detailed information about IPv6 AFT sessions on the device.

```
[Device] display aft session ipv6 verbose
Initiator:
  Source      IP/port: 2013::100/0
  Destination IP/port: 2012::1401:0101/32768
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 2012::1401:0101/0
  Destination IP/port: 2013::100/33024
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Local
State: ICMPV6_REPLY
Application: ICMP
Rule ID: -/-/-
Rule name:
Start time: 2014-03-13 08:52:59  TTL: 23s
Initiator->Responder:          4 packets       320 bytes
Responder->Initiator:          4 packets       320 bytes

Total sessions found: 1
```

# Display detailed information about IPv4 AFT sessions on the device.

```
[Device] display aft session ipv4 verbose
Initiator:
  Source      IP/port: 10.1.1.1/1025
  Destination IP/port: 20.1.1.1/2048
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Local
Responder:
  Source      IP/port: 20.1.1.1/1025
  Destination IP/port: 10.1.1.1/0
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Untrust
State: ICMP_REPLY
Application: ICMP
Rule ID: 0
Rule name: aftlocalout
Start time: 2014-03-13 08:52:59  TTL: 27s
Initiator->Responder:          4 packets       240 bytes
```

```
Responder->Initiator:          4 packets       240 bytes

Total sessions found: 1
```

Example: Providing FTP service from an IPv6 network to the IPv4 Internet

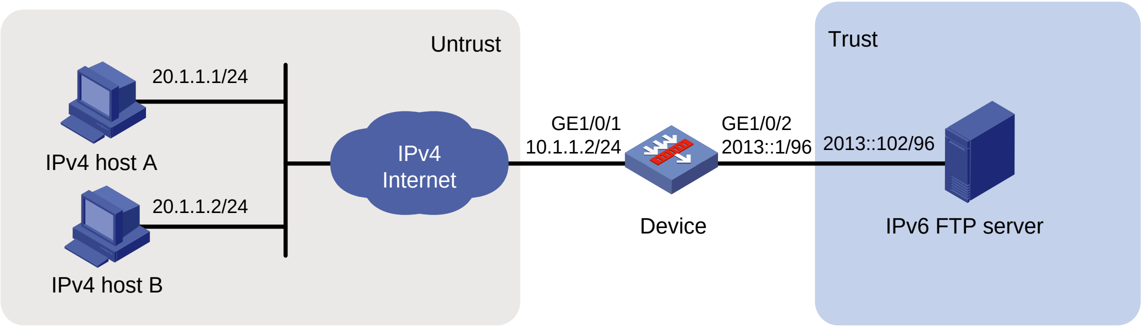
Network configuration

As shown in Figure 8, a company upgrades the network to IPv6, and it has an IPv4 address 10.1.1.1.

To allow the IPv6 FTP server to provide FTP services to IPv4 hosts, configure the following AFT policies on the device:

- Map the IPv6 FTP server's IPv6 address and TCP port number to the company's IPv4 address and TCP port number.
- Configure a NAT64 prefix to translate source IPv4 addresses of IPv4 packets to source IPv6 addresses.

Figure 8 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.1.1.2 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   ［Device］ security-zone name trust
   ［Device-security-zone-Trust］ import interface gigabitethernet 1/0/2
   ［Device-security-zone-Trust］ quit
   ［Device］ security-zone name untrust
   ［Device-security-zone-Untrust］ import interface gigabitethernet 1/0/1
   ［Device-security-zone-Untrust］ quit
   ```

3. Configure security policies:

   # In the IPv4 security policy, configure a rule named **aftlocalin** to allow the device to perform AFT on the IPv4 traffic destined for the IPv6 FTP server.

   ```
   [Device] security-policy ip
   ```

```
[Device-security-policy-ip] rule name aftlocalin
[Device-security-policy-ip-1-aftlocalin] source-zone untrust
[Device-security-policy-ip-1-aftlocalin] destination-zone local
[Device-security-policy-ip-1-aftlocalin] destination-ip-host 10.1.1.1
[Device-security-policy-ip-1-aftlocalin] action pass
[Device-security-policy-ip-1-aftlocalin] quit
[Device-security-policy-ip] quit
```

# In the IPv6 security policy, configure a rule named **aftlocalout** to allow the device to forward the AFT-translated packets to the IPv6 server.

```
[Device] security-policy ipv6
[Device-security-policy-ipv6] rule name aftlocalout
[Device-security-policy-ipv6-1-aftlocalout] source-zone local
[Device-security-policy-ipv6-1-aftlocalout] destination-zone trust
[Device-security-policy-ipv6-1-aftlocalout] source-ip-subnet 2012:: 96
[Device-security-policy-ipv6-1-aftlocalout] destination-ip-host 2013::102
[Device-security-policy-ipv6-1-aftlocalout] action pass
[Device-security-policy-ipv6-1-aftlocalout] quit
[Device-security-policy-ipv6] quit
```

4. Configure AFT settings:

# Map IPv4 address **10.1.1.1** with TCP port **21** to IPv6 address **2013::102** with TCP port **21** for the IPv6 internal FTP server.

```
[Device] aft v6server protocol tcp 10.1.1.1 21 2013::102 21
```

# Configure the device to use NAT64 prefix **2012:: 96** to translate source addresses of IPv4 packets.

```
[Device] aft prefix-nat64 2012:: 96
```

# Enable AFT on the interfaces connected to the IPv4 Internet and IPv6 network, respectively.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] aft enable
[Device-GigabitEthernet1/0/1] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] aft enable
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify that IPv4 hosts can use FTP to access the IPv6 FTP server. (Details not shown.)

# Display detailed information about IPv6 AFT sessions on the device.

```
[Device] display aft session ipv4 verbose
Initiator:
  Source     IP/port: 20.1.1.1/11025
  Destination IP/port: 10.1.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
Responder:
  Source     IP/port: 10.1.1.1/21
  Destination IP/port: 20.1.1.1/11025
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Local
State: TCP_ESTABLISHED
Application: FTP
Rule ID: -/-/-
Rule name:
Start time: 2014-03-13 09:07:30  TTL: 3577s
Initiator->Responder:          3 packets      124 bytes
Responder->Initiator:          2 packets      108 bytes

Total sessions found: 1
```

# Display detailed information about IPv4 AFT sessions on the device.

```
[Device] display aft session ipv6 verbose
Initiator:
  Source      IP/port: 2012::1401:0101/1029
  Destination IP/port: 2013::102/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Local
Responder:
  Source      IP/port: 2013::102/21
  Destination IP/port: 2012::1401:0101/1029
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
State: TCP_ESTABLISHED
Application: FTP
Rule ID: 0
Rule name: aftlocalout
Start time: 2014-03-13 09:07:30  TTL: 3582s
Initiator->Responder:          3 packets      184 bytes
Responder->Initiator:          2 packets      148 bytes

Total sessions found: 1
```

Example: Allowing mutual access between IPv4 and IPv6 networks

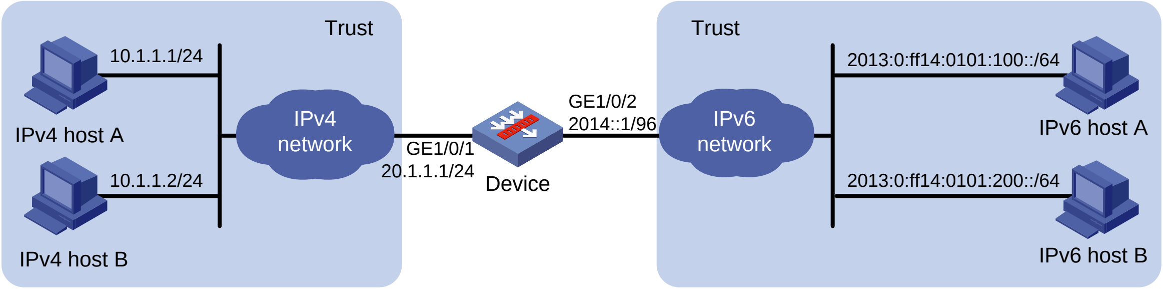
Network configuration

As shown in Figure 9, a company deploys both an IPv4 network and an IPv6 network.

To allow mutual access between the IPv4 network and the IPv6 network, configure the following AFT policies on the device:

- Assign an IVI prefix and an IPv4 subnet to the IPv6 network. Each IPv6 host uses the IPv6 addresses formed by the IVI prefix and an IPv4 address on the IPv4 subnet.
- Configure a NAT64 prefix to translate source IPv4 addresses of packets initiated by the IPv4 network to IPv6 addresses.

Figure 9 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 20.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

3. Configure settings for routing.

   This example configures an IPv6 static route (with next hop address 2014::100) and an IPv4 static route (with next hop address 20.1.1.2).

```
[Device] ipv6 route-static 2013:: 32 2014::100
[Device] ip route-static 10.1.1.0 24 20.1.1.2
```

4. Configure security policies to allow the device to permit IPv4-to-IPv6 traffic:

   # In the IPv4 security policy, configure a rule named **aftlocalin4** to allow the device to perform AFT on the IPv4 traffic destined for the IPv6 hosts.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name aftlocalin4
[Device-security-policy-ip-1-aftlocalin4] source-zone trust
[Device-security-policy-ip-1-aftlocalin4] destination-zone local
[Device-security-policy-ip-1-aftlocalin4] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-1-aftlocalin4] destination-ip-subnet 20.1.1.0 24
[Device-security-policy-ip-1-aftlocalin4] action pass
[Device-security-policy-ip-1-aftlocalin4] quit
[Device-security-policy-ip] quit
```

   # In the IPv6 security policy, configure a rule named **aftlocalout6** to allow the device to forward the AFT-translated packets to the IPv6 hosts.

```
[Device] security-policy ipv6
```

```
[Device-security-policy-ipv6] rule name aftlocalout6
[Device-security-policy-ipv6-1-aftlocalout6] source-zone local
[Device-security-policy-ipv6-1-aftlocalout6] destination-zone trust
[Device-security-policy-ipv6-1-aftlocalout6] source-ip-subnet 2012:: 96
[Device-security-policy-ipv6-1-aftlocalout6] destination-ip-subnet 2013:: 32
[Device-security-policy-ipv6-1-aftlocalout6] action pass
[Device-security-policy-ipv6-1-local-ipv6] quit
```

5. Configure security policies to allow the device to permit IPv6-to-IPv4 traffic:

   # In the IPv6 security policy, configure a rule named **aftlocalin6** to allow the device to perform AFT on the IPv6 traffic destined for the IPv4 hosts.

```
[Device-security-policy-ipv6] rule name aftlocalin6
[Device-security-policy-ipv6-2-aftlocalin6] source-zone trust
[Device-security-policy-ipv6-2-aftlocalin6] destination-zone local
[Device-security-policy-ipv6-2-aftlocalin6] source-ip-subnet 2013:: 32
[Device-security-policy-ipv6-2-aftlocalin6] destination-ip-subnet 2012:: 96
[Device-security-policy-ipv6-2-aftlocalin6] action pass
[Device-security-policy-ipv6-2-aftlocalin6] quit
[Device-security-policy-ipv6] quit
```

   # In the IPv4 security policy, configure a rule named **aftlocalout4** to allow the device to forward the AFT-translated packets to the IPv4 hosts.

```
[Device] security-policy ip
[Device-security-policy-ip] rule 2 name aftlocalout4
[Device-security-policy-ip-2-aftlocalout4] source-zone local
[Device-security-policy-ip-2-aftlocalout4] destination-zone trust
[Device-security-policy-ip-2-aftlocalout4] source-ip-subnet 20.1.1.0 24
[Device-security-policy-ip-2-aftlocalout4] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-2-aftlocalout4] action pass
[Device-security-policy-ip-2-aftlocalout4] quit
[Device-security-policy-ip] quit
```

6. Configure AFT settings:

   # Configure IPv4 ACL 2000 to permits all IPv4 packets to pass through.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit
[Device-acl-ipv4-basic-2000] quit
```

   # Configure the device to use NAT64 prefix **2012:: 96** to translate source addresses of IPv4 packets. The device also uses the prefix to translate destination addresses of IPv6 packets.

```
[Device] aft prefix-nat64 2012:: 96
```

   # Configure the device to use IVI prefix **2013::** to translate source addresses of IPv6 packets.

```
[Device] aft prefix-ivi 2013::
```

   # Configure the device to use IVI prefix **2013::** to translate destination addresses of packets permitted by IPv4 ACL 2000.

```
[Device] aft v4tov6 destination acl number 2000 prefix-ivi 2013::
```

   # Enable AFT on the interfaces connected to the IPv4 and IPv6 networks, respectively.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] aft enable
[Device-GigabitEthernet1/0/1] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] aft enable
```

```
    [Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify the connectivity between IPv6 hosts and IPv4 hosts. This example pings IPv4 host A from IPv6 host A.

```
D:\>ping 2012::a01:0101
Pinging 2012::a01:0101 with 32 bytes of data:
Reply from 2012::a01:0101: time=3ms
Reply from 2012::a01:0101: time=3ms
Reply from 2012::a01:0101: time=3ms
Reply from 2012::a01:0101: time=3ms
```

# Display information about IPv6 AFT sessions on the device.

```
[Device] display aft session ipv6 verbose
Initiator:
  Source     IP/port: 2013:0:FF14:0101:0100::/0
  Destination IP/port: 2012::0a01:0101/32768
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
Responder:
  Source     IP/port: 2012::0a01:0101/0
  Destination IP/port: 2013:0:FF14:0101:0100::/33024
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Local
State: ICMPV6_REPLY
Application: ICMP
Rule ID: -/-/-
Rule name:
Start time: 2014-03-13 08:52:59  TTL: 23s
Initiator->Responder:          4 packets       320 bytes
Responder->Initiator:          4 packets       320 bytes

Total sessions found: 1
```

# Display information about IPv4 AFT sessions on the device.

```
[Device] display aft session ipv4 verbose
Initiator:
  Source     IP/port: 20.1.1.1/1025
  Destination IP/port: 10.1.1.1/2048
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Local
Responder:
  Source     IP/port: 10.1.1.1/1025
  Destination IP/port: 20.1.1.1/0
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
State: ICMP_REPLY
Application: ICMP
Rule ID: 2
Rule name: aftlocalout4
Start time: 2014-03-13 08:52:59  TTL: 27s
Initiator->Responder:          4 packets       240 bytes
Responder->Initiator:          4 packets       240 bytes

Total sessions found: 1
```

Example: Allowing IPv6 Internet access from an IPv4 network

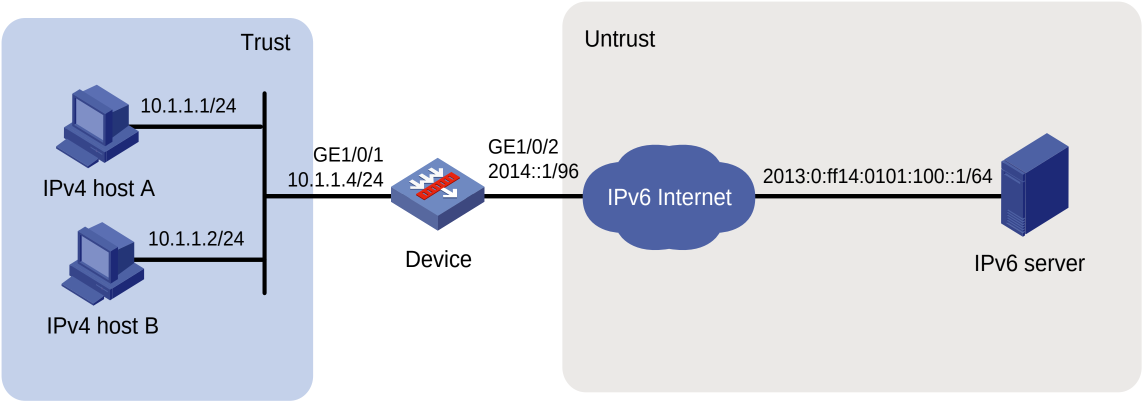
Network configuration

As shown in Figure 10, a company deploys an IPv4 network, and the Internet migrates to IPv6.

To allow IPv4 hosts to access the IPv6 server in the IPv6 Internet, configure the following AFT policies on the device:

- Configure an IPv4-to-IPv6 source address dynamic translation policy.
- Configure an IPv6-to-IPv4 source address static mapping for the IPv6 server.

Figure 10 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.1.1.4 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

3. Configure settings for routing.

   This example configures a static route with next hop address 2014::100.

```
[Device] ipv6 route-static 2013:0:ff14:0101:100:: 64 2014::100
```

4. Configure security policies:

   # In the IPv4 security policy, configure a rule named **aftlocalin** to allow the device to perform AFT on the IPv4 host traffic destined for the IPv6 server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name aftlocalin
[Device-security-policy-ip-1-aftlocalin] source-zone trust
[Device-security-policy-ip-1-aftlocalin] destination-zone local
[Device-security-policy-ip-1-aftlocalin] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-1-aftlocalin] destination-ip-host 20.1.1.1
[Device-security-policy-ip-1-aftlocalin] action pass
[Device-security-policy-ip-1-aftlocalin] quit
[Device-security-policy-ip] quit
```

   # In the IPv6 security policy, configure a rule named **aftlocalout** to allow the device to forward the AFT-translated packets to the IPv6 server.

```
[Device] security-policy ipv6
[Device-security-policy-ipv6] rule name aftlocalout
[Device-security-policy-ipv6-1-aftlocalout] source-zone local
[Device-security-policy-ipv6-1-aftlocalout] destination-zone untrust
[Device-security-policy-ipv6-1-aftlocalout] source-ip-subnet 2012:: 96
[Device-security-policy-ipv6-1-aftlocalout] destination-ip-host
2013:0:ff14:0101:100::1
[Device-security-policy-ipv6-1-aftlocalout] action pass
[Device-security-policy-ipv6-1-aftlocalout] quit
[Device-security-policy-ipv6] quit
```

5. Configure AFT settings:

   # Configure IPv4 ACL 2000 to permit IPv4 packets only from subnet 10.1.1.0/24 to pass through.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 10.1.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] rule deny
[Device-acl-ipv4-basic-2000] quit
```

   # Configure NAT64 prefix **2012:: 96**.

```
[Device] aft prefix-nat64 2012:: 96
```

   # Configure the device to use NAT64 prefix **2012:: 96** to translate source addresses of packets permitted by IPv4 ACL 2000.

```
[Device] aft v4tov6 source acl number 2000 prefix-nat64 2012:: 96
```

   # Map source IPv6 address **2013:0:ff14:0101:100::** to source IPv4 address **20.1.1.1**.

```
[Device] aft v6tov4 source 2013:0:ff14:0101:100:: 20.1.1.1
```

   # Enable AFT on the interfaces connected to the IPv4 and IPv6 networks, respectively.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] aft enable
[Device-GigabitEthernet1/0/1] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] aft enable
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify the connectivity between the IPv4 hosts and the IPv6 server. This example uses the ping utility on an IPv4 host.

```
D:\>ping 20.1.1.1
Pinging 20.1.1.1 with 32 bytes of data:
Reply from 20.1.1.1: bytes=32 time=14ms TTL=63
Reply from 20.1.1.1: bytes=32 time=1ms TTL=63
Reply from 20.1.1.1: bytes=32 time=1ms TTL=63
Reply from 20.1.1.1: bytes=32 time=1ms TTL=63
```

# Display detailed information about IPv6 AFT sessions on the device.

```
[Device] display aft session ipv4 verbose
Initiator:
  Source      IP/port: 10.1.1.1/1025
  Destination IP/port: 20.1.1.1/2048
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Trust
Responder:
  Source      IP/port: 20.1.1.1/1025
  Destination IP/port: 10.1.1.1/0
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Local
State: ICMP_REPLY
Application: ICMP
Rule ID: -/-/-
Rule name:
Start time: 2014-03-13 08:52:59  TTL: 27s
Initiator->Responder:          4 packets       240 bytes
Responder->Initiator:          4 packets       240 bytes

Total sessions found: 1
```

# Display detailed information about IPv4 AFT sessions on the device.

```
[Device] display aft session ipv6 verbose
Initiator:
  Source      IP/port: 2012::0A01:0101/0
  Destination IP/port: 2013:0:FF14:0101:0100::/32768
  VPN instance/VLAN ID/Inline ID: -/-/-
```

```
    Protocol: IPV6-ICMP(58)
    Inbound interface: GigabitEthernet1/0/1
    Source security zone: Local
  Responder:
    Source      IP/port: 2013:0:FF14:0101:0100::/0
    Destination IP/port: 2012::0A01:0101/33024
    VPN instance/VLAN ID/Inline ID: -/-/-
    Protocol: IPV6-ICMP(58)
    Inbound interface: GigabitEthernet1/0/2
    Source security zone: Untrust
  State: ICMPV6_REPLY
  Application: ICMP
  Rule ID: 0
  Rule name: aftlocalout
  Start time: 2014-03-13 08:52:59  TTL: 23s
  Initiator->Responder:          4 packets       320 bytes
  Responder->Initiator:          4 packets       320 bytes

  Total sessions found: 1
```

Example: Providing FTP service from an IPv4 network to the IPv6 Internet

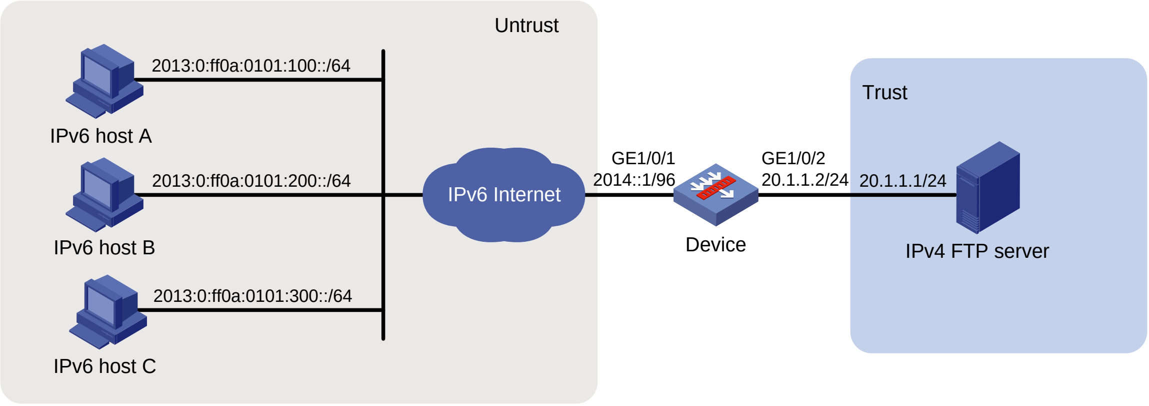
Network configuration

As shown in Figure 11, a company deploys an IPv4 network, and it has an IPv6 address 2012::1. The Internet migrates to IPv6.

To allow the IPv4 FTP server to provide FTP services to IPv6 hosts, configure the following AFT policies on the device:

- Configure an IPv4-to-IPv6 source address static mapping for the IPv4 FTP server. The device uses the mapping to translate the destination IPv6 address of IPv6-initiated addresses to the IPv4 address.
- Configure an IPv6-to-IPv4 source address dynamic translation policy. The device translates source IPv6 addresses of IPv6-initiated packets to source IPv4 addresses 30.1.1.1 and 30.1.1.2.

Figure 11 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.1.1.4 24
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] quit
   ```

3. Configure security policies:

   # In the IPv6 security policy, configure a rule named **aftlocalin** to allow the device to perform AFT on the IPv6 host traffic destined for the IPv4 server.

   ```
   [Device] security-policy ipv6
   [Device-security-policy-ipv6] rule name aftlocalin
   [Device-security-policy-ipv6-1-aftlocalin] source-zone untrust
   [Device-security-policy-ipv6-1-aftlocalin] destination-zone local
   [Device-security-policy-ipv6-1-aftlocalin] destination-ip-host 2012::1
   [Device-security-policy-ipv6-1-aftlocalin] action pass
   [Device-security-policy-ipv6-1-aftlocalin] quit
   [Device-security-policy-ipv6] quit
   ```

   # In the IPv4 security policy, configure a rule named **aftlocalout** to allow the device to forward the AFT-translated packets to the IPv4 server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name aftlocalout
   [Device-security-policy-ip-1-aftlocalout] source-zone local
   [Device-security-policy-ip-1-aftlocalout] destination-zone trust
   [Device-security-policy-ip-1-aftlocalout] source-ip-host 30.1.1.1
   [Device-security-policy-ip-1-aftlocalout] source-ip-host 30.1.1.2
   [Device-security-policy-ip-1-aftlocalout] destination-ip-host 20.1.1.1
   [Device-security-policy-ip-1-aftlocalout] action pass
   [Device-security-policy-ip-1-aftlocalout] quit
   ```

4. Configure AFT settings:

   # Map source IPv4 address **20.1.1.1** to source IPv6 address **2012::1**.

   ```
   [Device] aft v4tov6 source 20.1.1.1 2012::1
   ```

   # Configure address group 0, and add the address range from 30.1.1.1 to 30.1.1.2 to the group.

   ```
   [Device] aft address-group 0
   [Device-aft-address-group-0] address 30.1.1.1 30.1.1.2
   [Device-aft-address-group-0] quit
   ```

   # Configure IPv6 ACL 2000 to permit all IPv6 packets to pass through.

   ```
   [Device] acl ipv6 basic 2000
   [Device-acl-ipv6-basic-2000] rule permit
   ```

```
[Device-acl-ipv6-basic-2000] quit
```

# Configure the device to translate source addresses of IPv6 packets permitted by IPv6 ACL 2000 to IPv4 addresses in address group 0.

```
[Device] aft v6tov4 source acl ipv6 number 2000 address-group 0
```

# Enable AFT on the interfaces connected to the IPv6 Internet and IPv4 network, respectively.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] aft enable
[Device-GigabitEthernet1/0/1] quit
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] aft enable
[Device-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Verify the connectivity between the IPv6 hosts and the IPv4 FTP server. For example, ping the IPv4 FTP server from IPv6 host A.

```
D:\>ping 2012::1
Pinging 2012::1 with 32 bytes of data:
Reply from 2012::1: time=3ms
Reply from 2012::1: time=3ms
Reply from 2012::1: time=3ms
Reply from 2012::1: time=3ms
```

# Display detailed information about IPv6 AFT sessions on the device.

```
[Device] display aft session ipv6 verbose
Initiator:
  Source      IP/port: 2013:0:FF0A:0101:0100::/1029
  Destination IP/port: 2012::1/21
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
Responder:
  Source      IP/port: 2012::1/21
  Destination IP/port: 2013:0:FF0A:0101:0100::/1029
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: IPV6-ICMP(58)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Local
State: ICMPV6_REPLY
Application: ICMP
Rule ID: -/-/-
Rule name:
Start time: 2014-03-13 09:07:30  TTL: 3582s
Initiator->Responder:          3 packets       184 bytes
Responder->Initiator:          2 packets       148 bytes

Total sessions found: 1
```

# Display detailed information about IPv4 AFT sessions on the device.

```
[Device] display aft session ipv4 verbose
Initiator:
  Source      IP/port: 30.1.1.1/11025
  Destination IP/port: 20.1.1.1/21
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Local
Responder:
  Source      IP/port: 20.1.1.1/21
  Destination IP/port: 30.1.1.1/11025
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: ICMP(1)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
State: ICMP_REPLY
Application: ICMP
Rule ID: 0
Rule name: aftlocalout
Start time: 2014-03-13 09:07:30  TTL: 3577s
Initiator->Responder:          3 packets       124 bytes
Responder->Initiator:          2 packets       108 bytes

Total sessions found: 1
```

NSFOCUS Firewall Series

NF VPN Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for VPN features, including:SSL VPN, IPsec, tunneling, GRE, L2TP, and ADVPN.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

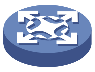
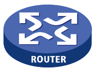
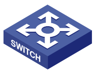
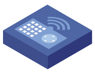
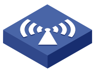
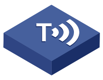
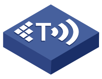
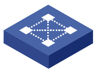
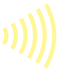
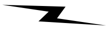
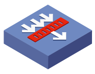
GUI conventions

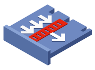
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring SSL VPN

About SSL VPN

SSL VPN provides SSL-based secure remote access services through an SSL VPN gateway. Users from anywhere on the Internet can establish a secure connection to an SSL VPN gateway through an SSL-enabled browser to access protected resources behind the gateway.

SSL VPN operating mechanism

To allow remote user access to protected resources behind an SSL VPN gateway, you must configure these resources on the gateway. Remote users can access only the resources authorized to them after they establish an SSL-encrypted connection to the gateway and pass the identity authentication.

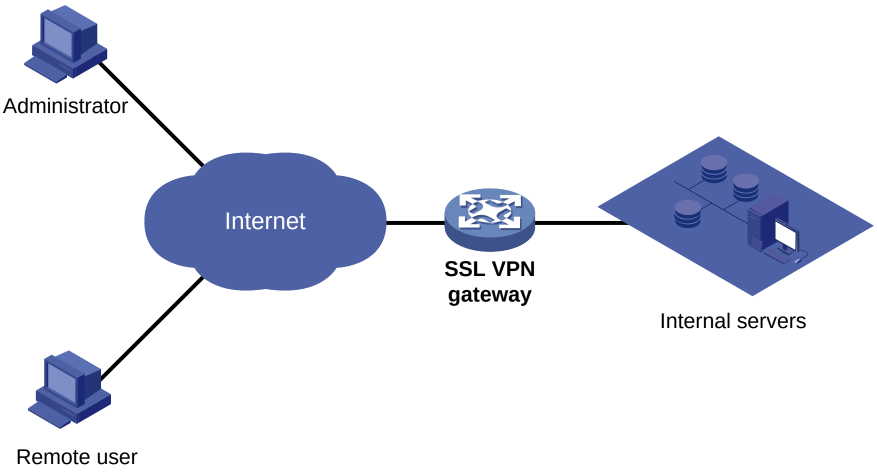
As shown in Figure 1, SSL VPN operates as follows:

1. The remote user establishes an HTTPS connection to the SSL VPN gateway.

   In this process, the remote user and the SSL VPN gateway perform SSL certificate authentication.
2. The remote user enters the username and password.
3. The SSL VPN gateway authenticates the credentials that the user entered, and authorizes the user to access a range of resources.
4. The user selects a resource to access.

   An access request for that resource is sent to the SSL VPN gateway through the SSL connection.
5. The SSL VPN gateway resolves the request and forwards the request to the corresponding internal server.
6. The SSL VPN gateway forwards the server's reply to the user through the SSL connection.

Figure 1 SSL VPN network diagram

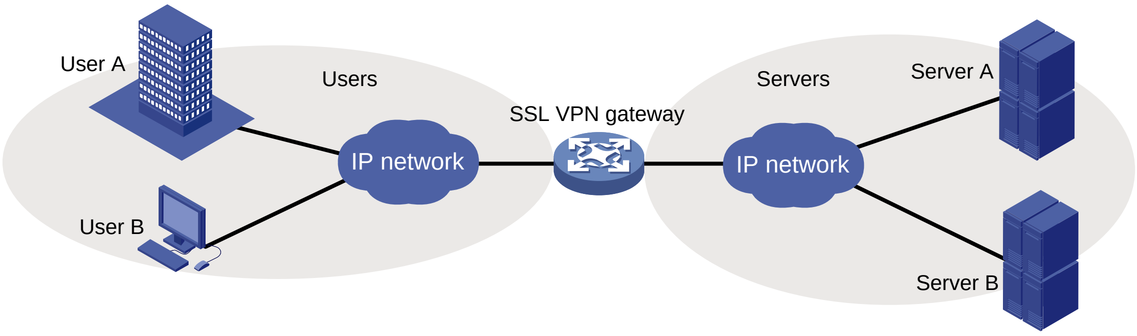


SSL VPN networking modes

Gateway mode

In gateway mode, the SSL VPN gateway acts as a gateway that connects remote users and the internal servers network, as shown in Figure 2. Because the SSL VPN gateway is deployed in line, it can provide full protection to the internal network but it affects data transmission performance.

Figure 2 Gateway mode

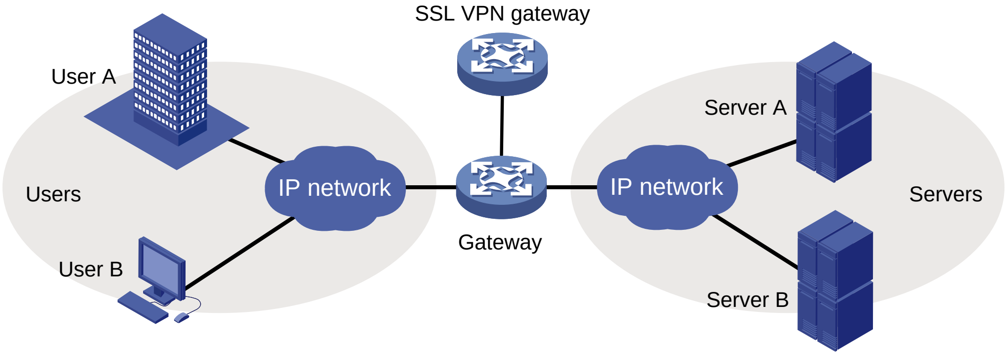


Single-arm mode

In single-arm mode, the SSL VPN gateway is attached to the network gateway, as shown in Figure 3.

The gateway forwards user-to-server traffic to the SSL VPN gateway. The SSL VPN gateway processes the traffic and sends the processed traffic back to the gateway. The gateway forwards the traffic to the internal servers. The SSL VPN gateway is not a performance bottleneck in the network because it is not deployed on the key path. However, the SSL VPN gateway cannot provide full protection to the internal network.

Figure 3 Single-arm mode



SSL VPN access modes

Web access

In Web access mode, remote users use browsers to access Web resources allowed by an SSL VPN gateway through HTTPS. After login, a user can access any resources listed on the webpage. In Web access mode, all operations are performed on webpages.

The resources available for SSL VPN Web access users are Web servers only.

To implement Web access, you must configure a list of URLs on the SSL VPN gateway. A URL is the IP address or domain name of an internal Web server.

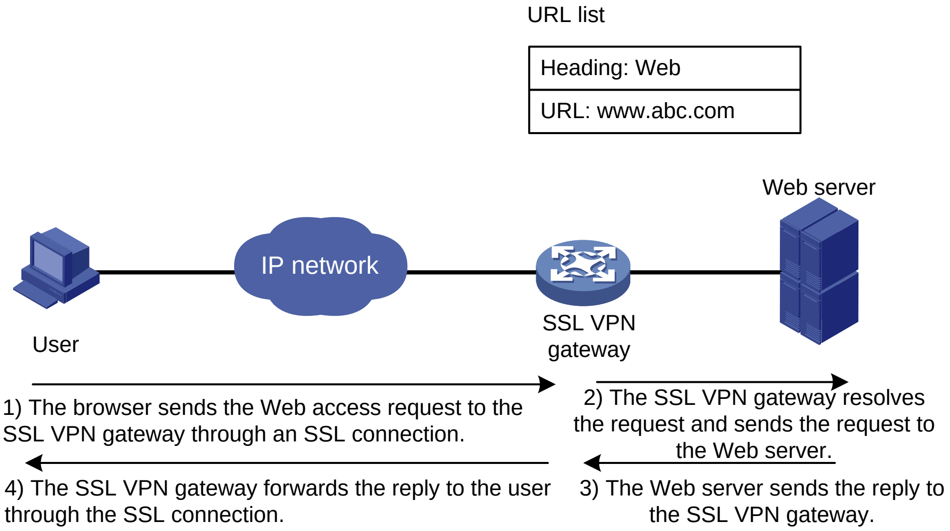
The Web access procedure is as follows:

1. A user uses a browser to log in to an SSL VPN gateway through HTTPS.
2. The SSL VPN gateway authenticates the user and authorizes the user to access the available URLs.

   The authorized URLs are displayed on the SSL VPN gateway webpage as URL links.
3. The user selects a URL to access on the SSL VPN gateway webpage. The browser sends the access request to the SSL VPN gateway through the SSL connection for HTTPS.
4. The SSL VPN gateway resolves the request and sends the request to the Web server through HTTP or HTTPS.
5. After receiving the reply from the Web server, the SSL VPN gateway forwards the reply to the user through the SSL connection for HTTPS.

Figure 4 illustrates the Web access process. The administrator configures a URL of www.nsfocus.com.cn on the SSL VPN gateway. Then, the SSL VPN user can access the internal Web server by accessing the URL on the SSL VPN gateway webpage.

Figure 4 Network diagram for Web access


URL list
Heading: Web
URL: www.abc.com
Web server
IP network
User
SSL VPN gateway
1) The browser sends the Web access request to the SSL VPN gateway through an SSL connection.
2) The SSL VPN gateway resolves the request and sends the request to the Web server.
4) The SSL VPN gateway forwards the reply to the user through the SSL connection.
3) The Web server sends the reply to the SSL VPN gateway.

TCP access

In TCP access mode, users access TCP applications on internal servers by accessing the applications' open ports. Supported applications include remote access services (such as Telnet), desktop sharing services, mail services, Notes services, and other TCP services that use fixed ports.

In TCP access mode, a user installs the TCP access client software on the SSL VPN client (the terminal device that the user uses). The client software uses an SSL connection to transmit the application layer data.

To implement TCP access, you must configure port forwarding instances on the SSL VPN gateway. A port forwarding instance maps a TCP service (identified by an IP address/domain name and port number) to an SSL VPN client's local IP address (or host name) and port number.

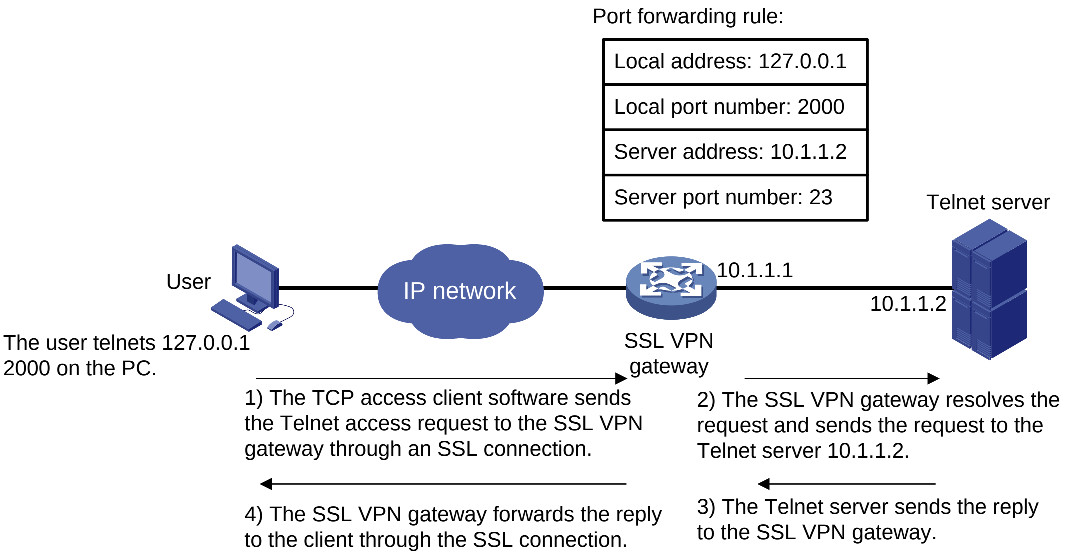
The TCP access procedure is as follows:

1. A user uses a browser to log in to an SSL VPN gateway through HTTPS.
2. The SSL VPN gateway authenticates the user and authorizes the user to access the Telnet service (port forwarding instance).
3. The user downloads the TCP access client software from the webpage of the SSL VPN gateway, and launches the software. The software opens the authorized local port in the port forwarding instance.
4. The user tries to access the local IP address and port number. The TCP access client software sends the access request to the SSL VPN gateway through an SSL connection.

5. The SSL VPN gateway resolves the request and sends the request to the Telnet server according to the port forwarding instance.
6. After receiving the reply from the Telnet server, the SSL VPN gateway forwards the reply to the user through the SSL connection.

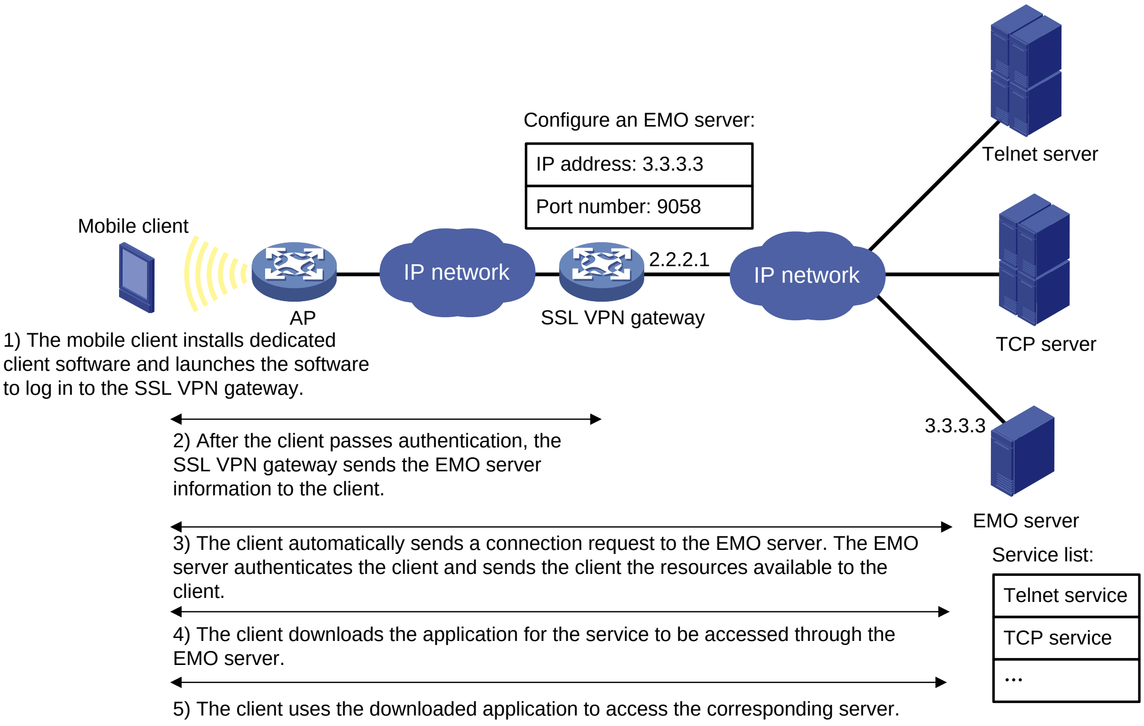
As shown in Figure 5, the administrator creates a port forwarding instance for the Telnet service on the SSL VPN gateway. The rule maps the internal Telnet server address 10.1.1.2 and port number 23 to the SSL VPN client's local address 127.0.0.1 and local port number 2000. Then, the SSL VPN user can access the internal Telnet server by telneting the local address 127.0.0.1 and local port number 2000.

Figure 5 Network diagram for TCP access


Port forwarding rule:
Local address: 127.0.0.1
Local port number: 2000
Server address: 10.1.1.2
Server port number: 23
Telnet server
User
IP network
10.1.1.1
10.1.1.2
SSL VPN gateway
The user telnets 127.0.0.1 2000 on the PC.
1) The TCP access client software sends the Telnet access request to the SSL VPN gateway through an SSL connection.
2) The SSL VPN gateway resolves the request and sends the request to the Telnet server 10.1.1.2.
3) The Telnet server sends the reply to the SSL VPN gateway.
4) The SSL VPN gateway forwards the reply to the client through the SSL connection.

For mobile clients to use the TCP access mode, you do not need to configure port forwarding instances on the SSL VPN gateway. However, client software dedicated for mobile clients is required, and you must specify an Endpoint Mobile Office (EMO) server for mobile clients on the SSL VPN gateway. Mobile clients access internal resources through the EMO server. Figure 6 shows the access process.

Figure 6 Network diagram for mobile client access to internal servers


Configure an EMO server:
IP address: 3.3.3.3
Port number: 9058
Telnet server
Mobile client
2.2.2.1
IP network
IP network
AP
SSL VPN gateway
TCP server
1) The mobile client installs dedicated client software and launches the software to log in to the SSL VPN gateway.
3.3.3.3
2) After the client passes authentication, the SSL VPN gateway sends the EMO server information to the client.
EMO server
3) The client automatically sends a connection request to the EMO server. The EMO server authenticates the client and sends the client the resources available to the client.
Service list:
Telnet service
TCP service
...
4) The client downloads the application for the service to be accessed through the EMO server.
5) The client uses the downloaded application to access the corresponding server.

IP access

IP access implements secured IP communication between remote users and internal servers.

To access an internal server in IP access mode, a user must install dedicated IP access client software. The client software will install a virtual network interface card (VNIC) on the SSL VPN client.

To implement IP access, you must configure the following on the SSL VPN gateway:

- An SSL VPN AC interface.
- Routes to accessible IP resources. The routes will be issued to SSL VPN clients to instruct packet forwarding.

Figure 7 uses a ping operation to illustrate the IP access process. The administrator must first configure a route to the ping destination (server 10.1.1.2/24) on the SSL VPN gateway.

The access process is as follows:

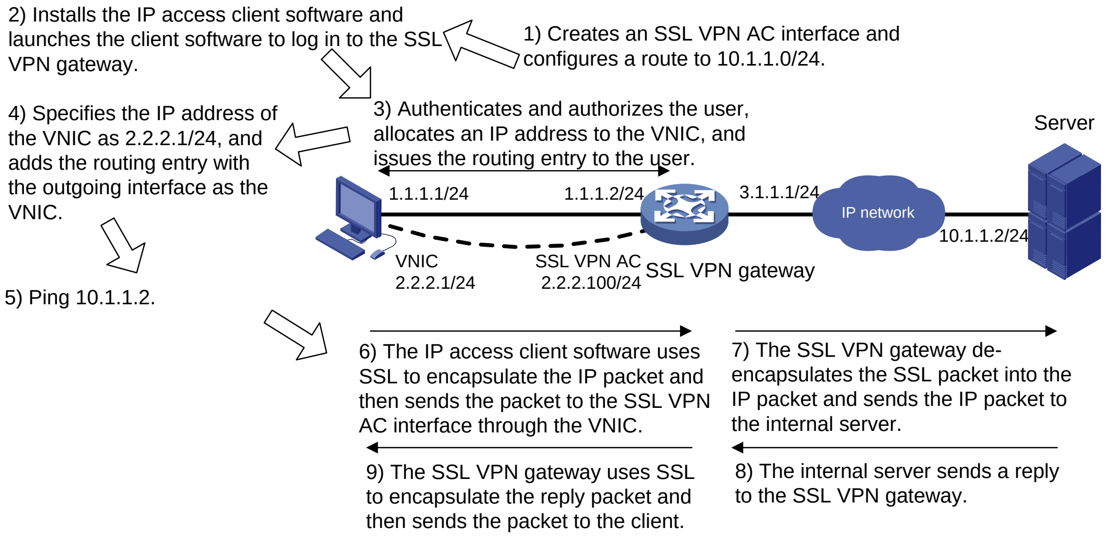
1. The user installs the IP access client software and launches the client software to log in to the SSL VPN gateway.
2. The SSL VPN gateway performs the following operations:
   a. Authenticates and authorizes the user.
   b. Allocates an IP address to the VNIC of the user.
   c. Issues the authorized IP access resources to the client.

      In this example, a route to server 10.1.1.2/24 is issued.
3. The client specifies the allocated IP address as the VNIC's address and adds the route to the local routing table, using the VNIC as output interface.
4. The user pings the server address.

   The ping request matches the route. Matching packets will be encapsulated by SSL.
5. The client uses SSL to encapsulate the ping request packet, and then sends the packet to the SSL VPN AC interface through the VNIC.

6. The SSL VPN gateway de-encapsulates the SSL packet into the IP packet and forwards the IP packet to the corresponding internal server.
7. The internal server sends a reply to the SSL VPN gateway.
8. The SSL VPN gateway uses SSL to encapsulate the reply packet and then sends the packet to the client through the SSL VPN AC interface.

Figure 7 Network diagram for IP access


2) Installs the IP access client software and launches the client software to log in to the SSL VPN gateway.
1) Creates an SSL VPN AC interface and configures a route to 10.1.1.0/24.
4) Specifies the IP address of the VNIC as 2.2.2.1/24, and adds the routing entry with the outgoing interface as the VNIC.
3) Authenticates and authorizes the user, allocates an IP address to the VNIC, and issues the routing entry to the user.
Server
1.1.1.1/24
1.1.1.2/24
3.1.1.1/24
IP network
10.1.1.2/24
VNIC
2.2.2.1/24
SSL VPN AC
2.2.2.100/24
SSL VPN gateway
5) Ping 10.1.1.2.
6) The IP access client software uses SSL to encapsulate the IP packet and then sends the packet to the SSL VPN AC interface through the VNIC.
7) The SSL VPN gateway de-encapsulates the SSL packet into the IP packet and sends the IP packet to the internal server.
9) The SSL VPN gateway uses SSL to encapsulate the reply packet and then sends the packet to the client.
8) The internal server sends a reply to the SSL VPN gateway.

SSL VPN user authentication

To access resources in an SSL VPN context, a user must first pass identity authentication to log in to the SSL VPN context. The authentication methods for an SSL VPN context include username/password authentication, certificate authentication, verification code authentication, SMS authentication, and custom authentication.

If custom authentication and SMS authentication are enabled at the same time, only custom authentication takes effect. You can enable username/password authentication, certificate authentication, or both in an SSL VPN context. Whether these authentication methods are required for logging in to the SSL VPN context depends on the configuration of the **authentication use** command. To use username/password authentication for users, you must also create accounts for the users in AAA. For more information, see "Configuring AAA."

You can also enable the verification code authentication, SMS authentication, and custom authentication in an SSL VPN context. These authentication methods are required for login authentication if they are configured.

Username/password authentication

The username/password authentication process is as follows:

1. The SSL VPN user enters the login username and password on the SSL VPN login page. The username and password are sent to the SSL VPN gateway.
2. The SSL VPN gateway sends the received username and password to AAA for authentication, authorization, and accounting, or to a custom authentication server for authentication and authorization.

Certification authentication

As shown in Figure 8, the certificate authentication process is as follows:

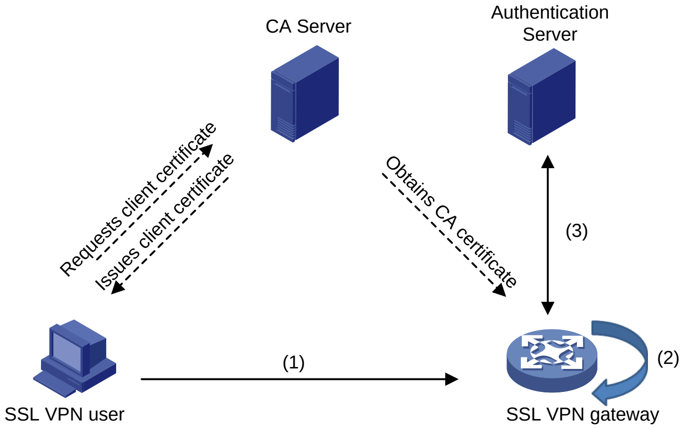
1. The SSL VPN user selects the certificate for login when prompted. The certificate is sent in an SSL connection request to the SSL VPN gateway.
2. The SSL VPN gateway verifies the validity of the user certificate.
   - If the certificate is verified as invalid, the gateway rejects the SSL connection request. The user cannot log in to the SSL VPN context.
   - If the certificate is verified as valid, the SSL connection is established and the gateway performs the next step.
3. The SSL VPN gateway checks for certificate revocation if CRL checking is enabled.
   - If the certificate is verified as not revoked, the SSL connection is established and the gateway performs the next step.
   - If the certificate is verified as revoked, the gateway rejects the SSL connection request. The user cannot log in to the SSL VPN context.

   For more inforamtion about CRL checking, see "Configuring PKI."
4. The SSL VPN gateway extracts the username from the certificate attribute (CN attribute by default). Then, the SSL VPN gateway sends the username to AAA for authorization and accounting, or to a custom authentication server for authorization.

NOTE:

To use certificate authentication, make sure the username extracted from the specified certificate attribute exists on the authentication server.

Figure 8 Certificate authentication process



Combined username/password authentication and certificate authentication

The authentication process of combined username/password authentication and certificate authentication is as follows:

1. The SSL VPN user selects the certificate for login when prompted. The certificate is sent in an SSL connection request to the SSL VPN gateway.
2. The SSL VPN gateway verifies the validity of the user certificate.
   - If the certificate is verified as invalid, the gateway rejects the SSL connection request. The user cannot log in to the SSL VPN context.
   - If the certificate is verified as valid, the SSL connection is established and the gateway performs the next step.
3. The SSL VPN gateway checks for certificate revocation if CRL checking is enabled.
   - If the certificate is verified as not revoked, the SSL connection is established and the gateway performs the next step.

    - If the certificate is verified as revoked, the gateway rejects the SSL connection request. The user cannot log in to the SSL VPN context.
4. The SSL VPN gateway extracts the username from the certificate and compares the extracted username with the username provided by the user:
    - The user passes identity authentication if the two usernames match. The SSL VPN gateway then sends the username and password to AAA for authentication, authorization and accounting, or to a custom authentication server for authentication and authorization.
    - The user fails the identity authentication if the two usernames do not match.

NOTE:

A user might enter the username and password when the user selects the certificate or after the SSL connection is established, depending on the access mode.

SMS authentication

After you enable SMS authentication, the device uses SMS verification codes to authenticate SSL VPN users. A user is allowed to log in to the SSL VPN gateway only when the user passes the SMS authentication.

The device supports the following types of SMS authentication:

- IMC SMS authentication.

  SMS authentication for SSL VPN users is performed by an IMC server. You must configure the IP address and port number for the IMC server in IMC SMS authentication view.
- SMS gateway authentication.

  SMS gateway authentication for SSL VPN users is performed by an SMS gateway. You must specify the SMS gateway, the verification code resend interval, and the verification code validity period in SMS gateway authentication view.

The two SMS authentication types cannot both be configured.

For SMS gateway authentication, one username can be bound to only one mobile number. When multiple users log in to the SSL VPN gateway by using the same username, the users must check the verification codes reception order. A user must submit the verification code received right for his own login attempt.

Custom authentication

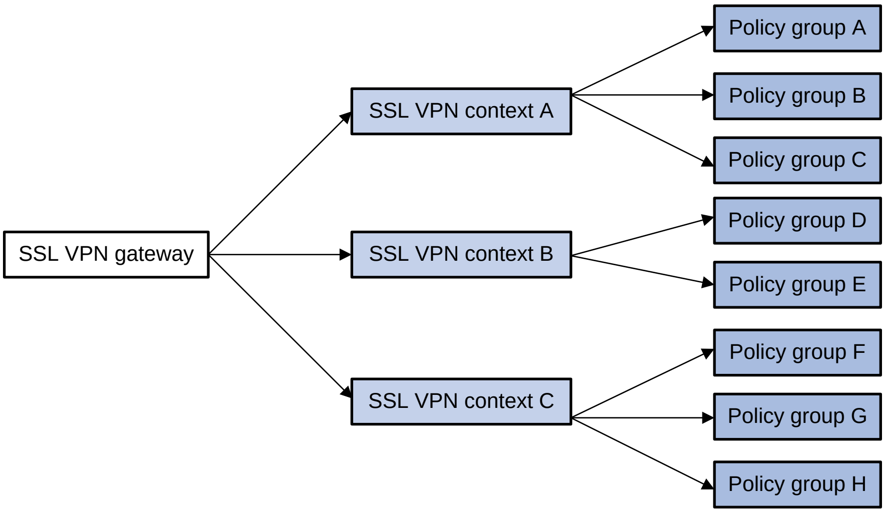
Custom authentication allows you to set up and configure a custom authentication server as needed. The device can use the custom authentication server for user authentication and authorization. The custom authentication server does not support accounting.

Resource access control

SSL VPN controls user access to resources on a per-user basis.

As shown in Figure 9, an SSL VPN gateway can be associated with multiple SSL VPN contexts. An SSL VPN context contains multiple policy groups. A policy group defines accessible Web resources, TCP resources, and IP resources.

Figure 9 SSL VPN resource access control



You can specify domain names or virtual host names for the SSL VPN contexts associated with an SSL VPN gateway. When a user logs in to the SSL VPN gateway, the SSL VPN gateway performs the following operations:

1. Uses the domain name or virtual host name that the user entered to determine the SSL VPN context to which the user belongs.
2. Uses the authentication and authorization methods of the ISP domain specified for the context to perform authentication and authorization for the user.
    - If the SSL VPN gateway authorizes the user to use a policy group, the user can access resources allowed by the policy group.
    - If the SSL VPN gateway does not authorize the user to use a policy group, the user can access resources allowed by the default policy group.

NOTE:

The SSL VPN gateway uses an AAA server or a custom authentication server to perform user authentication and authorization. SSL VPN supports AAA protocols RADIUS and LDAP. RADIUS is most often used.

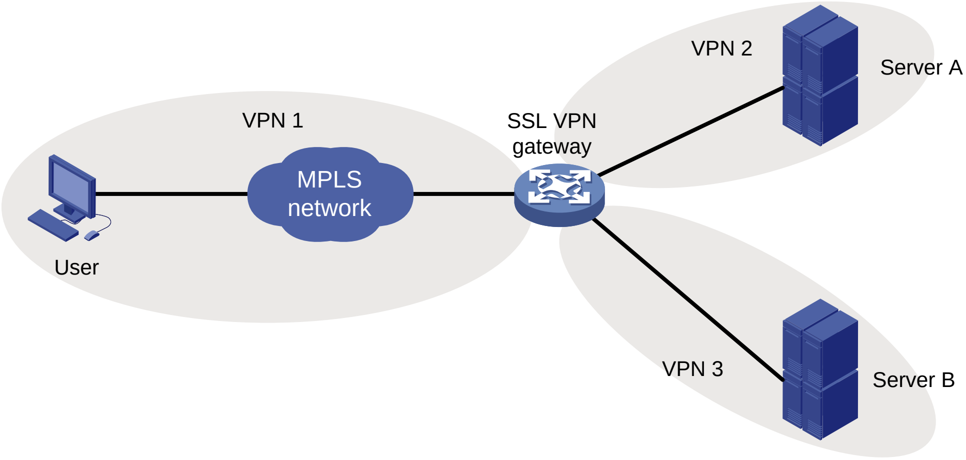
VRF-aware SSL VPN

VRF-aware SSL VPN provides the following functionalities:

- **VRF-aware SSL VPN context**—You associate different SSL VPN contexts with different VRF instances (VPN instances) on the SSL VPN gateway. Users in an SSL VPN context can access only the resources in the VPN instance associated with the SSL VPN context. VRF-aware SSL VPN contexts also allow server addresses to overlap.
- **VRF-aware SSL VPN gateway**—You specify the VPN instance to which the SSL VPN gateway belongs. Only users in the same VPN can access the SSL VPN gateway. The VRF-aware SSL VPN gateway prevents the internal server resources from leaking into the public network or other VPNs.

For more information about VPN instances, see VPN instance configuration in *VPN Instance Configuration Guide*.

Figure 10 VRF-aware SSL VPN



Restrictions: Licensing requirements for SSL VPN

The following compatibility matrixes show the maximum number of online user accounts supported by default:

Models	Default max online user accounts
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	125

You can purchase and install a license to increase the number of supported online users. For more information about licenses, see license management in *Fundamentals Configuration Guide*.

In an IRF network, the maximum number of online users supported by an IRF fabric is calculated as follows: Maximum online users supported by the IRF fabric = Sum of the maximum online users permitted by the license of each member device + maximum online users supported by default.

After a member device becomes faulty, its license can still take effect on the IRF fabric for 60 days. As a best practice, purchase and install a license for each IRF member device.

In other HA networks, a license for a member device takes effect only on the member device itself. For other member devices to use SSL VPN after the licensed member device becomes faulty, you must purchase and install a license for each member device.

Restrictions and guidelines: SSL VPN configuration

The SSL VPN gateway generates only one session for a user who accesses both Web and IP resources in the following method:

1. First, the user accesses the SSL VPN gateway through a Web browser.
2. Then, the user downloads the IP access client through the Web page and launches the IP access client.

Once the user exits the Web browser or IP access client, the session is terminated and the user can access neither Web nor IP access resources.

You can specify ACLs for user access filtering in an SSL VPN policy group. Rules in the specified ACLs do not take effect if they contain VPN settings.

In a browser-SSL VPN gateway-server network, the gateway cannot process the redirect URL carried with a reply from the internal server because the gateway cannot edit request or reply packets.

SSL VPN tasks at a glance

To configure SSL VPN, perform the following tasks on the SSL VPN gateway:

1. Configuring an SSL VPN gateway
2. Configuring an SSL VPN context
3. Configuring SSL VPN user authentication, authorization, and accounting
   a. Configuring user authentication in an SSL VPN context
   b. Configuring the SSL VPN user authentication server

      A custom authentication server must be configured for custom authentication.
4. Configuring SSL VPN resource access control as needed
   - Configuring a URI ACL
   - Configuring the Web access service
   - Configuring the TCP access service
   - Configuring the IP access service
   - Configuring SSL VPN access for mobile clients
   - (Optional.) Configuring shortcuts
   - (Optional.) Configuring redirect resources
   - (Optional.) Configuring HTTP redirection
   - (Optional.) Configuring the default policy group for an SSL VPN context
5. (Optional.) Configuring VRF-aware SSL VPN
   - Associating an SSL VPN context with a VPN instance
   - Specifying a VPN instance for an SSL VPN gateway
6. (Optional.) Configuring SSL VPN user control
   - Configuring online SSL VPN user control
   - Configuring SSL VPN session rate limit
   - Configuring SSL VPN cracking prevention
   - Configuring SSL VPN SSO login
   - Configuring WeChat Work authentication
7. (Optional.) Customizing SSL VPN webpages
   - Customizing SSL VPN webpage elements
   - Specifying an SSL VPN webpage template
8. (Optional.) Enabling SSL VPN logging

Prerequisites for SSL VPN

Before you configure the SSL VPN gateway, complete the following tasks:

- Configure PKI and obtain a digital certificate for the SSL VPN gateway (see "Configuring PKI").

- Configure an SSL server policy to be used by the SSL VPN gateway (see "Configuring SSL").

Configuring an SSL VPN gateway

Restrictions and guidelines

An SSL VPN gateway that uses the default IPv4 or IPv6 address must use a port number that is different from the HTTPS service port number.

If the settings of the SSL server policy applied to an SSL VPN gateway are changed, you must disable and then enable the SSL VPN gateway to use the modified policy.

The IP address and port number of an SSL VPN gateway cannot both be the same as those of the HTTPS server on the device. Otherwise, you can access only the SSL VPN Web interface but cannot access the device management Web interface by using those IP address and port number.

An SSL VPN gateway can use an IPv4 address, an IPv6 address, but not both. If you configure both IPv4 and IPv6 addresses, the most recent configuration takes effect.

When a remote user uses a browser to establish an HTTPS connection with the SSL VPN gateway, the gateway supports providing two types of local certificates to the peer. These local certificates include self-signed certificates signed by the device and local certificates signed by a CA for the SSL VPN gateway. This enables remote users to perform digital certificate-based authentication on the SSL VPN gateway. The remote user can choose a type of certificate according to security requirements and configuration complexity as follows:

- **Self-signed certificate**—Using this type of certificate is easy in configuration but has low insecure. You do not need to associate an SSL server policy with the SSL VPN gateway and the default SSL settings are used. A self-signed certificate is not assigned by a CA and therefore not trusted by the browser of the remote user, so the browser will prompt a security risk. If the remote user does not have high security requirements and can accept the security risk, the remote user can ignore the prompt and continue browsing the webpage.
- **Local certificate signed by a CA for the SSL VPN gateway**—Using this type of certificate is complex in configuration but has high security. To use this type of certificate, you must perform the following tasks:
  - Obtain a CA certificate and request a local certificate from the CA.
  - Specify an SSL server policy for the SSL VPN gateway.

For more information about digital certificates and SSL server policies, see "Configuring PKI" and "Configuring SSL", respectively.

To improve packet transmission security, a server policy supports only TLS 1.1 or higher for SSL negotiation by default. To log in to the iNode client, make sure the iNode client supports TLS 1.1 or higher. To resolve this issue, you can upgrade the iNode to the latest version.

Procedure

1. Enter system view.

   system-view

2. Create an SSL VPN gateway and enter its view.

   sslvpn gateway *gateway-name*

3. Configure an IPv4 address and a port number for the SSL VPN gateway.

   ip address *ip-address* [**port** *port-number*]

   By default, the SSL VPN gateway uses IPv4 address 0.0.0.0 and port number 443.

   If you configure the **ip address** command without specifying a port number, the default port number (443) is used.

4. Configure an IPv6 address and a port number for the SSL VPN gateway.

   ipv6 address *ipv6-address* [**port** *port-number*]

By default, no IPv6 address or port number is configured for the SSL VPN gateway.

If you configure the **ipv6 address** command without specifying a port number, the default port number (443) is used.

5. Apply an SSL server policy to the SSL VPN gateway.

   ssl server-policy *policy-name*

   By default, an SSL VPN gateway uses the SSL server policy of its self-signed certificate.

6. Enable the SSL VPN gateway.

   service enable

   By default, the SSL VPN gateway is disabled.

Configuring an SSL VPN context

About this task

An SSL VPN context manages user sessions and resources available to users.

Restrictions and guidelines

When you associate an SSL VPN context with an SSL VPN gateway, follow these guidelines:

- Make sure the context has a domain name or virtual host name different than any existing contexts associated with the SSL VPN gateway.
- If you do not specify a domain name or virtual host name for the context, you cannot associate other SSL VPN contexts with the SSL VPN gateway.
- If you specify a virtual host name, deploy a DNS server in the network to resolve the virtual host name to the SSL VPN gateway's IP address.
- Typically, a virtual host name for the context cannot contain underscores (_). For SSL VPN Web access, if a virtual host name contains underscores (_), the virtual host name will be added as part of domian names. When you specify an SSL VPN gateway and a virtual host for the context and the name of the virtual host contains underscores (_), the IE browser cannot save cookies and will not carry cookies in requests, which will cause an SSL VPN login failure.

You can associate an SSL VPN context with a maximum of 10 SSL VPN gateways.

Procedure

1. Enter system view.

   system-view

2. Create an SSL VPN context and enter its view.

   sslvpn context *context-name*

3. Associate the context with an SSL VPN gateway.

   gateway *gateway-name* [**domain** *domain-name* | **virtual-host** *virtual-host-name*]

   By default, the context is not associated with an SSL VPN gateway.

4. Specify an ISP domain for AAA of SSL VPN users in the context.

   aaa domain *domain-name*

   By default, the default ISP domain is used for AAA of SSL VPN users in an SSL VPN context.

   An SSL VPN username cannot carry ISP domain information. After this command is executed, the SSL VPN gateway uses the specified domain for AAA of SSL VPN users in the context.

5. Enable the context.

   service enable

   By default, the context is disabled.

6. (Optional.) Set the maximum number of sessions (online users) for the context.

   max-users *max-number*

   By default, an SSL VPN context supports a maximum of 1048575 sessions (online users).

7. (Optional.) Set the idle timeout timer for SSL VPN sessions.

   timeout idle *minutes*

   By default, the idle timeout timer for SSL VPN sessions is 30 minutes.

8. (Optional.) Set the idle-cut traffic threshold for SSL VPN sessions.

   idle-cut traffic-threshold

   By default, the SSL VPN session idle-cut traffic threshold is 0 bytes. An SSL VPN session will be disconnected if no traffic is transmitted within the session idle timeout time specified by the **timeout idle** command.

9. (Optional.) Apply an SSL client policy to the SSL VPN context.

   ssl client-policy *policy-name*

   The default SSL client policy for SSL VPN is used. This policy supports the **dhe_rsa_aes_128_cbc_sha**, **dhe_rsa_aes_256_cbc_sha**, **rsa_3des_ede_cbc_sha**, **rsa_aes_128_cbc_sha**, and **rsa_aes_256_cbc_sha** cipher suites.

   The SSL VPN gateway will use the settings in the specified SSL client policy to connect to HTTPS servers.

10. (Optional.) Enable URL masking globally.

    url-masking enable

    URL masking is disabled by default.

    After URL masking is enabled, the URLs of the Web access resources configured in the SSL VPN context are converted into coded strings.

Configuring user authentication in an SSL VPN context

Restrictions and guidelines for user authentication configuration in an SSL VPN context

How certificate authentication works depends on the configuration of the **client-verify** command in SSL server policy view. You can use the command to enable mandatory or optional SSL client authentication. Mandatory certificate authentication is supported only for Web users and IP access users. For TCP access users and mobile client users to access the SSL VPN gateway successfully, optional SSL client authentication must be used.

User authentication tasks at a glance

To configure user authentication in an SSL VPN context, perform the following tasks:

1. Specifying the authentication methods required for user login
2. Configuring basic authentication methods
   - Configuring username/password authentication
   - Configuring certificate authentication
3. (Optional.) Configuring verification code authentication
4. (Optional.) Configuring IMC SMS authentication
5. (Optional.) Configuring SMS gateway authentication

6. (Optional.) Configuring password modification for users

Specifying the authentication methods required for user login

About this task

You can enable username/password authentication, certificate authentication, or both in an SSL VPN context. Whether these authentication methods are required for logging in to the SSL VPN context depend on the configuration of the **authentication use all** command:

- If the **authentication use all** command is configured, a user must pass all the enabled authentication methods for login.
- If the **authentication use any-one** command is configured, a user can log in after passing any enabled authentication method.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Specify the authentication methods required for user login.

   authentication use { **all** | **any-one** }

   By default, a user must pass all the enabled authentication methods to log in to an SSL VPN context.

Configuring username/password authentication

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable username/password authentication.

   password-authentication enable

   Username/password authentication is enabled by default.

Configuring certificate authentication

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable certificate authentication.

   certificate-authentication enable

   Certificate authentication is disabled by default.

4. Specify the certificate attribute as the SSL VPN username.

   certificate username-attribute { **cn** | **email-prefix** | **oid** *extern-id* }

   By default, the device uses the value of the CN attribute in the subject of the certificate as the SSL VPN username.

Configuring verification code authentication

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable verification code authentication.

   verify-code enable

   By default, verification code authentication is enabled.

Configuring IMC SMS authentication

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable IMC SMS authentication.

   sms-auth type imc

   By default, IMC SMS authentication is disabled.

4. Create and enter IMC SMS authentication view.

   sms-auth imc

5. Specify an IMC server.

   server-address *ip-address* **port** *port-number* [**vpn-instance** *vpn-instance-name*]

   By default, no IMC server is specified.

Configuring SMS gateway authentication

Prerequisites

Complete the SMS gateway configuration. For information about configuring an SMS gateway, see "Configuring SMS."

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enter SSL VPN user view.

   user *username*

4. Specify the mobile number for the SSL VPN user to receive SMS messages.

   mobile-num *number*

   By default, no mobile number is specified for receiving SMS messages.

5. Return to SSL VPN context view.

   quit

6. Enable SMS gateway authentication.

sms-auth type sms-gw

By default, SMS gateway authentication is disabled.

7. Create and enter SMS gateway authentication view.

   sms-auth sms-gw

   By default, the SMS gateway authentication view does not exist.

8. Specify an SMS gateway.

   gateway *sms-gateway-name*

   By default, no SMS gateway is specified.

9. Enable mobile number binding.

   mobile-num-binding enable

   By default, mobile number binding is disabled.

10. Set the verification code resend interval.

    verification-code send-interval *seconds*

    By default, the verification code resend interval is 60 seconds.

11. Set the verification code validity period.

    verification-code validity *minutes*

    By default, the verification code validity period is one minute.

12. Specify the mobile country code.

    country-code *country-code*

    By default, the mobile country code is 86.

13. Configure the SMS content template.

    sms-content *string*

    By default, the SMS content template is **Hello, $$USER$$, the verification code is $$VERIFYCODE$$, and its validity period is $$VALIDTIME$$ in minutes.**.

Configuring password modification for users

About this task

Password modification allows SSL VPN users to modify login passwords on the personal settings page after logging in to the SSL VPN Web interface. This feature is available only for IMC authentication users.

If you disable this feature, the modify password function will be hidden on the SSL VPN Web interface, so users cannot modify their passwords.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable SSL VPN users in the SSL VPN context to modify passwords.

   password-changing enable

   By default, SSL VPN users in the SSL VPN context are allowed to modify passwords.

4. Enter SSL VPN user view.

   user *username*

5. (Optional.) Enable password modification for the SSL VPN user.

```
password-changing enable
```

By default, an SSL VPN user is allowed to modify the password.

6. Specify an IMC server for password modification.

   ```
   self-service imc address ip-address port port-number [ vpn-instance vpn-instance-name ]
   ```

   By default, no IMC server is specified for password modification.

   Execute this command only when IMC authentication users need to modify the SSL VPN login passwords.

Configuring the SSL VPN user authentication server

Specifying the SSL VPN user authentication server type

About this task

The SSL VPN user authentication supports the following types of servers:

- **AAA authentication server**—The device uses an AAA server for user authentication, authorization, and accounting. For more information about AAA, see "Configuring AAA."
- **Custom authentication server**—You can set up and configure a custom authentication server as needed. The device can use the custom authentication server for user authentication and authorization. The custom authentication server does not support accounting. For more information about configuring the custom authentication server, see "Configuring the custom authentication server."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter SSL VPN context view.

   ```
   sslvpn context context-name
   ```

3. Specify the authentication server type.

   ```
   authentication server-type { aaa | custom }
   ```

   By default, the SSL VPN authentication server is an AAA server.

Configuring the custom authentication server

About this task

To use a custom authentication server for user authentication and authorization, configure the following settings:

- URL of the custom authentication server.

  The SSL VPN gateway uses HTTP to send authentication requests to the specified URL.
- Custom authentication timeout.

  After sending an HTTP request to the custom authentication server, the SSL VPN gateway waits for reponses from the server. If the gateway receives no response within the authentication timeout, it returns an authentication failure message to the SSL VPN client.
- HTTP request settings for custom authentication.

The SSL VPN gateway constructs an HTTP request based on the authentication request settings, including HTTP request method, request header fields, and request template.

- HTTP response settings for custom authentication.

  The SSL VPN gateway parses an HTTP response based on the authentication response settings. The settings include HTTP response format, authentication success value in the response, field names in the response, and response templates for the custom-format HTTP response.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Configure the URL of the custom authentication server.

   custom-authentication url *url*

   By default, no custom authentication server URL is configured.

4. Specify the custom authentication timeout.

   custom-authentication timeout *seconds*

   By default, the custom authentication timeout is 15 seconds.

5. Configure settings for a custom authentication request:

   a. Configure the HTTP request method.

      custom-authentication request-method { **get** | **post** }

      By default, the HTTP request method is GET.

   b. Configure HTTP request header fields.

      custom-authentication request-header-field *field-name* **value** *value*

      By default, a custom authentication request header includes the following fields:

      – **Content-type:application/x-www-form-urlencoded**.

      – **User-Agent:nodejs 4.1**.

      – **Accept:text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q**.

   c. Configure the HTTP request template.

      custom-authentication request-template *template*

      By default, no request template is configured.

6. Configure settings for a custom authentication response:

   a. Specify the HTTP response format.

      custom-authentication response-format { **custom** | **json** | **xml** }

      By default, the HTTP response format is JSON.

   b. Configure the authentication success value in the HTTP response.

      custom-authentication response-success-value *success-value*

      By default, no authentication success value is configured for the HTTP response.

   c. Configure field names in the HTTP response.

      custom-authentication response-field { **group** *group* | **message** *message* | **result** *result* }

      By default, no HTTP response field names are configured.

      You must configure HTTP response field names if the HTTP response format is JSON or XML.

   d. Configure response templates for the fields in the custom-format HTTP response.

custom-authentication response-custom-template { **group** | **message** | **result** } *template*

By default, no response templates are configured.

The response templates are required when the HTTP response format is custom.

Configuring a URI ACL

About this task

A URI ACL is a set of rules that permit or deny access to resources. You can use URI ACLs for fine-grained IP, TCP, and Web access filtering of SSL VPN users.

You can add multiple rules to a URI ACL. The device matches a packet against the rules in ascending order of rule ID. The match process stops once a matching rule is found.

You can create multiple URI ACLs in an SSL VPN context.

A URI ACL can filter SSL VPN users' HTTP, HTTPS, TCP, UDP, ICMP, and IP traffic based on the following fields:

- Protocol type.
- IP address.
- Host name.
- Port number.
- URL.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a URI ACL and enter its view.

   uri-acl *uri-acl-name*

4. Configure a rule in the URI ACL.

   rule [*rule-id*] { **deny** | **permit** } **uri** *uri-pattern-string*

   By default, no rules are configured in a URI ACL.

Configuring the Web access service

To allow remote users to access internal resources in Web access mode, you must configure Web access resources and associate the resources with an SSL VPN policy group.

Restrictions and guidelines

A webpage compatibility error might occur for Web access. As a best practice, use IP access.

Web access service tasks at a glance

To configure the Web access service, perform the following tasks:

1. Configuring a URL list
2. Configuring an SSL VPN policy group for Web access

3. (Optional.) Configuring a file policy

Configuring a URL list

About this task

A URL list is a list of URL items that define the accessible Web resources behind the SSL VPN gateway. Each URL item corresponds to an internal Web resource.

The SSL VPN gateway rewrites the resource URL returned from the internal server before sending the URL to the requesting user. The URL mapping type determines how the gateway rewrites the URL.

The following example describes how URL mapping works when the user accesses internal resources at URL **http://www.server.com:8080**. The SSL VPN gateway name is **gw**, domain name is **https://www.gateway.com:4430**, and IP address is **1.1.1.1**.

- **Normal rewriting**—This is the default mapping method. The resource URL returned to the client will be rewritten to **https://www.gateway.com:4430/_proxy2/http/8080/www.server.com**.
- **Domain mapping**—The resource URL returned to the client will be rewritten to **https://***mapped domain name***:4430**, where *mapped domain name* is the user-defined domain name.
- **Port mapping**—You can specify a gateway name with or without a virtual host name for port mapping. For example:
    - If you specify **gw2** as the gateway name and do not specify a virtual host name, the resource URL will be rewritten to **https://2.2.2.2:4430**, where 2.2.2.2 and 4430 are the IP address and port number of SSL VPN gateway **gw2**.
    - If you specify **gw** as the gateway name and **vhosta** as the virtual host name, the resource URL will be rewritten to https://vhosta:4430.

Restrictions and guidelines

Resource URL rewriting is available only for resource access responses that contain HTML, XML, CSS, or JavaScript files.

Normal rewriting might cause problems such as missed URL rewriting and rewriting errors, resulting in SSL VPN clients not being able to access the internal resources. Use domain mapping or URL mapping as a best practice.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a URL item and enter its view.

   url-item *name*

4. Specify the resource URL in the URL item.

   url *url*

   By default, no resource URL is specified in a URL item.

   If you do not specify a protocol type in the resource URL, the default protocol (HTTP) is used.

5. (Optional.) Enable URL masking.

   url-masking enable

   By default, URL masking is disabled.

   After URL masking is enabled, the Web resource URL for the URL item is converted into a coded string.

6. (Optional.) Specify a URI ACL in the URL item.

   resources uri-acl *uri-acl-name*

   By default, no URI ACL is specified.

7. (Optional.) Configure the URL mapping method.

   url-mapping { **domain-mapping** *domain-name* | **port-mapping gateway** *gateway-name* [**virtual-host** *virtual-host-name*] } [**rewrite-enable**]

   By default, the normal rewriting method is used.

8. Return to SSL VPN context view.

   quit

9. Create a URL list and enter its view.

   url-list *name*

10. (Optional.) Configure a heading for the URL list.

    heading *string*

    By default, the URL list heading is **Web**.

11. Add the URL item to the URL list.

    resources url-item *name*

    By default, a URL list does not contain any URL items.

Configuring an SSL VPN policy group for Web access

About this task

To configure an SSL VPN policy group for Web access, associate a URL list with the policy group. After the authentication server authorizes a user to use a policy group, the user can access the Web resources provided by the URL list associated with the policy group.

In a policy group, you can specify an advanced ACL and a URI ACL to filter users' Web access requests.

The advanced ACL supports filtering Web access requests by destination IP address and destination port number. The URI ACL supports filtering Web access requests by protocol type, destination address, domain name, port number, and URL.

The SSL VPN gateway uses the following procedure to determine whether to forward a Web access request:

1. Matches the request against the authorized URL list.
   - If the request matches a URL item in the list, the gateway forwards the request.
   - If the request does not match any URL items in the list, the gateway proceeds to the next step.
2. Matches the request against rules in the URI ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the URI ACL or if no URI ACL is available, the gateway proceeds to the next step.
3. Matches the request against rules in the advanced ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the advanced ACL or if no advanced ACL is available, the gateway drops the request.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create an SSL VPN policy group and enter SSL VPN policy group view.

   policy-group *group-name*

4. Associate a URL list with the policy group.

   resources url-list *url-list-name*

   By default, no URL list associated with a policy group.

5. (Optional.) Specify the ACLs for Web access filtering:
   - Specify an advanced ACL for Web access filtering.

     filter web-access [**ipv6**] **acl** *advanced-acl-number*
   - Specify a URI ACL for Web access filtering.

     filter web-access uri-acl *uri-acl-name*

   By default, users can access only the Web resources authorized to them through the URL list.

Configuring a file policy

About this task

A file policy enables the SSL VPN gateway to rewrite Web page files before forwarding them to requesting Web access users.

A file policy contains the following settings:

- A URL that identifies the path of the file to which the file policy is applied.
- One or more rewrite rules.

  A rewrite rule defines the old file content to be rewritten and the new content used to replace the old content.
- (Optional.) The file type that the file is changed to after being rewritten by the file policy.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a file policy and enter its view.

   file-policy *policy-name*

   By default, no file policies exist.

4. Specify the URL of the file to be rewritten.

   url *url*

   By default, no file URL is specified in a file policy.

5. Specify the file type that a file is changed to after being rewritten by the file policy.

   content-type { **css** | **html** | **javascript** | **other** }

   By default, a file policy rewrites a file in an HTTP response to the file type indicated by the content-type field in the HTTP response.

6. Create a rewrite rule and enter its view.

rewrite-rule *rule-name*

7. Specify the old content to be rewritten.

   old-content *string*

   By default, the old content to be rewritten is not specified.

8. Specify the new content used to replace the old content.

   new-content *string*

   By default, the new content used to replace the old content is not specified.

Configuring the TCP access service

To allow remote users to access internal resources in TCP access mode, you must configure TCP access resources and associate the resources with an SSL VPN policy group.

TCP access service tasks at a glance

To configure the TCP access service, perform the following tasks:

1. Configuring a port forwarding list
2. Configuring an SSL VPN policy group for TCP access

Configuring a port forwarding list

About this task

A port forwarding list is a list of port forwarding items. Each port forwarding item contains a port forwarding instance.

A port forwarding instance maps a TCP service (such as Telnet, SSH, or POP3) hosted on an internal server to a local address and port number on the SSL VPN client. Remote users can access the TCP service though the local address and port number.

The port forwarding instance is displayed together with the port forwarding item name on the SSL VPN Web page. If you configure a resource link for the port forwarding item, the port forwarding item name will be displayed as a link on the SSL VPN Web page. You can click the link to access the resource directly.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a port forwarding item and enter its view.

   port-forward-item *item-name*

4. Configure a port forwarding instance for the port forwarding item.

   local-port *local-port-number* **local-name** *local-name* **remote-server** *remote-server* **remote-port** *remote-port-number* [**description** *text*]

5. (Optional.) Configure a resource link for the port forwarding item.

   execution *script*

6. Return to SSL VPN context view.

   quit

7. Create a port forwarding list and enter its view.

```
port-forward port-forward-name
```

8. Assign the port forwarding item to the port forwarding list.

   ```
   resources port-forward-item item-name
   ```

   By default, a port forwarding list does not contain port forwarding items.

Configuring an SSL VPN policy group for TCP access

About this task

To configure an SSL VPN policy group for TCP access, associate a port forwarding list with the policy group. After the authentication server authorizes a user to use a policy group, the user can access the TCP services provided by the port forwarding list associated with the policy group.

In a policy group, you can specify an advanced ACL and a URI ACL to filter users' TCP access requests.

The advanced ACL supports filtering TCP access requests by destination IP address and destination port number. The URI ACL supports filtering TCP access requests by protocol type, destination address, domain name, port number, and URL.

For PC users, the ACLs configured for TCP access filtering do not take effect. They can access only the TCP resources authorized to them through the TCP port forwarding list.

For mobile client users, the SSL VPN gateway uses the following procedure to determine whether to forward a TCP access request:

1. Matches the request against the authorized port forwarding list.
   - If the request matches a port forwarding item in the list, the gateway forwards the request.
   - If the request does not match any port forwarding items in the list, the gateway proceeds to the next step.
2. Matches the request against the rules in the URI ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the URI ACL or if no URI ACL is available, the gateway proceeds to the next step.
3. Matches the request against the rules in the advanced ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the advanced ACL or if no advanced ACL is available, the gateway drops the request.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter SSL VPN context view.

   ```
   sslvpn context context-name
   ```

3. Create an SSL VPN policy group and enter SSL VPN policy group view.

   ```
   policy-group group-name
   ```

4. Associate a port forwarding list with the policy group.

   ```
   resources port-forward port-forward-name
   ```

   By default, no port forwarding list is associated with a policy group.

5. (Optional.) Specify the ACLs for TCP access filtering:

- Specify an advanced ACL for TCP access filtering.

  filter tcp-access [**ipv6**] **acl** *advanced-acl-number*

- Specify a URI ACL for TCP access filtering.

  filter tcp-access uri-acl *uri-acl-name*

By default, users can access only the TCP resources authorized to them through the TCP port forwarding list.

Configuring the IP access service

To allow remote users to access internal resources in IP access mode, you must configure IP access resources and associate the resources with an SSL VPN policy group.

Restrictions and guidelines for IP access service configuration

To ensure correct forwarding of reply packets to an SSL VPN client, configure static routes from the internal servers to the network segment where the client's VNIC resides.

IP access service tasks at a glance

To configure the IP access service, perform the following tasks:

1. Configuring an SSL VPN AC interface for IP access
2. Creating an address pool for IP access users
3. Configuring IP access parameters in an SSL VPN context
4. Configuring an SSL VPN policy group for IP access
5. (Optional.) Binding IP addresses to an SSL VPN user

Configuring an SSL VPN AC interface for IP access

Configuring an SSL VPN AC interface

1. Enter system view.

   system-view

2. Create an SSL VPN AC interface and enter its view.

   interface sslvpn-ac *interface-number*

3. Configure an IP address for the interface.

   ip address *ip-address* { *mask* | *mask-length* }

   By default, no IP address is configured for an AC interface.

4. (Optional.) Set the expected bandwidth for the interface.

   bandwidth *bandwidth-value*

   The expected bandwidth is 64 kbps by default.

   The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

5. (Optional.) Configure the description of the interface.

   description *text*

   The default interface description is *interface name* **Interface**. For example, **SSLVPN-AC1000 Interface**.

6. (Optional.) Set the MTU of the interface.

   mtu *size*

   The default MTU is 1500 bytes.

7. Bring up the interface.

   undo shutdown

   By default, an SSL VPN AC interface is up.

Restoring the default settings for the SSL VPN AC interface

> **IMPORTANT:**
>
> Restoring the default interface settings might interrupt ongoing network services. Make sure you are fully aware of the impact of this operation when you perform it on a live network.

To restore the default settings for the SSL VPN AC interface:

1. Enter system view.

   system-view

2. Enter SSL VPN AC interface view.

   interface sslvpn-ac *interface-number*

3. Restore the default settings for the SSL VPN AC interface.

   default

   This command might fail to restore the default settings for some commands for reasons such as command dependencies and system restrictions. You can use the **display this** command in interface view to check for these commands, and use their **undo** forms or follow the command reference to restore their respective default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Creating an address pool for IP access users

About this task

Create an IP address pool for the SSL VPN gateway to assign IP addresses to the VNICs used by IP access users.

Restrictions and guidelines

To prevent IP address conflicts, make sure the IP addresses in the address pool meet the following requirements:

- Not in the same network segment as the physical NICs on the clients.
- Exclude IP addresses of the interfaces on the device that acts the SSL VPN gateway.
- Not in the same network segment as the internal addresses to be accessed.

Procedure

1. Enter system view.

   system-view

2. Create an address pool.

   sslvpn ip address-pool *pool-name start-ip-address end-ip-address*

Configuring IP access parameters in an SSL VPN context

About this task

To provide service to IP access users, you must configure IP access parameters in an SSL VPN context, including the SSL VPN AC interface, address pool, and route list. After a user passes identity authentication, the SSL VPN context allocates an IP address to the VNIC of the user from the specified address pool. The route list can be used by an SSL VPN policy group to issue route entries to users.

Restrictions and guidelines

Automatic pushing of accessible resources to IP access users through the Web page is available only for users that use the iNode client in Windows. You can install the iNode client by using one of the following methods:

- Log in to the SSL VPN gateway from a Web browser, and then download and install the iNode client that comes with the device.
- Install the iNode client downloaded from the official website. Select the iNode installation package for VPN gateway generation when customizing the iNode client. If you do not select this option, the user will be automatically logged out because the SSL VPN gateway cannot detect that the iNode client is logged in.

Procedure

1. Enter system view.

   `system-view`

2. Enter SSL VPN context view.

   `sslvpn context` *`context-name`*

3. Specify an SSL VPN AC interface for IP access.

   `ip-tunnel interface sslvpn-ac` *`interface-number`*

   By default, no SSL VPN AC interface is specified for IP access in the SSL VPN context.

4. Configure a route list:

   a. Create a route list and enter its view.

   `ip-route-list` *`list-name`*

   b. Add an included route to the route list.

   `include` *`ip-address`* { *`mask`* | *`mask-length`* }

   c. Add an excluded route to the route list.

   `exclude` *`ip-address`* { *`mask`* | *`mask-length`* }

   d. Return to SSL VPN context view.

   `quit`

5. Specify an address pool for IP access.

   `ip-tunnel address-pool` *`pool-name`* **`mask`** { *`mask-length`* | *`mask`* }

   By default, no address pool is specified for IP access.

6. (Optional.) Set the keepalive interval.

   `ip-tunnel keepalive` *`seconds`*

   By default, the keepalive interval is 30 seconds.

7. (Optional.) Specify a DNS server for IP access.

   `ip-tunnel dns-server` { **`primary`** | **`secondary`** } *`ip-address`*

   By default, no DNS servers are specified for IP access.

8. (Optional.) Specify a WINS server for IP access.

ip-tunnel wins-server { **primary** | **secondary** } *ip-address*

By default, no WINS servers are specified for IP access.

9. (Optional.) Enable automatic startup of the IP access client after Web login.

web-access ip-client auto-activate

By default, automatic startup of the IP access client after Web login is disabled.

10. (Optional.) Enable automatic pushing of accessible resources to IP access users through the Web page.

ip-tunnel web-resource auto-push

By default, automatic pushing of accessible resources to IP access users through the Web page is disabled.

11. (Optional.) Set a rate limit for IP access upstream or downstream traffic.

ip-tunnel rate-limit { **downstream** | **upstream** } { **kbps** | **pps** } *value*

By default, no rate limit is set for IP access upstream or downstream traffic.

Configuring an SSL VPN policy group for IP access

About this task

To configure an SSL VPN policy group for IP access, configure routes for the accessible IP resources in the policy group. After the AAA server or custom authentication server authorizes a user to use a policy group, the SSL VPN gateway issues the routes to the user so the user can access the IP resources.

You can configure the routes to be issued to users by using one of the following methods:

- Manually configure a route.
- Specify a route list.
- Force all traffic to be sent to the SSL VPN gateway.

  The SSL VPN gateway issues a default route to the SSL VPN client. The default route uses the VNIC as the output interface and has the highest priority among all default routes on the client. Packets for destinations not in the routing table are sent to the SSL VPN gateway through the VNIC. The SSL VPN gateway monitors the SSL VPN client in real time. It does not allow the client to delete the default route or add a default route with a higher priority.

In a policy group, you can specify an advanced ACL and a URI ACL to filter users' IP access requests.

The SSL VPN gateway uses the following procedure to determine whether to forward an IP access request:

1. Matches the request against the rules in the URI ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the URI ACL or if no URI ACL is available, the gateway proceeds to step 2.
2. Matches the request against the rules in the advanced ACL:
   - If the request matches a permit rule, the gateway forwards the request.
   - If the request matches a deny rule, the gateway drops the request.
   - If the request does not match any rules in the advanced ACL or if no advanced ACL is available, the gateway drops the request.

If no URI ACL or advanced ACL is specified for IP access filtering, the SSL VPN gateway permits all IP accesses by default.

The advanced ACL supports filtering IP access requests by using the following criteria:

- Destination IP address.
- Destination port number.
- Source IP address.
- Source port number.
- Protocol type.
- Packet priority.
- Fragment information.
- TCP flag.
- ICMP message type and message code.

The URI ACL supports filtering IP access requests by protocol type, destination address, domain name, port number, and URL.

Restrictions and guidelines

If a rule in the URI ACL specified for IP access filtering contains HTTP or HTTPS settings, the rule does not take effect.

Procedure

1. Enter system view.

   `system-view`

2. Enter SSL VPN context view.

   `sslvpn context` *`context-name`*

3. Create an SSL VPN policy group and enter SSL VPN policy group view.

   `policy-group` *`group-name`*

4. Specify the routes to be issued to clients.

   `ip-tunnel access-route` { *`ip-address`* { *`mask-length`* | *`mask`* } | **`force-all`** | **`ip-route-list`** *`list-name`* }

   By default, no routes are configured.

5. Specify the ACLs for IP access filtering:

   - Specify an advanced ACL for IP access filtering.

     `filter ip-tunnel` [**`ipv6`**] **`acl`** *`advanced-acl-number`*

   - Specify a URI ACL for IP access filtering.

     `filter ip-tunnel uri-acl` *`uri-acl-name`*

   By default, an SSL VPN gateway permits all IP access requests.

6. (Optional.) Specify an address pool for IP access.

   `ip-tunnel address-pool` *`pool-name`* **`mask`** { *`mask-length`* | *`mask`* }

   By default, no address pool is specified for IP access in an SSL VPN policy group.

   If no free address is available in the address pool or the address pool does not exist, address allocation to IP access users will fail and the users' access requests will be rejected.

   If no address pool is specified for the policy group, the SSL VPN gateway allocates IP addresses to users from the address pool specified for the SSL VPN context.

Binding IP addresses to an SSL VPN user

About this task

When an SSL VPN user accesses the SSL VPN gateway in IP access mode, the SSL VPN gateway must assign an IP address to the user. This feature allows you to specify the IP addresses that can be assigned to a user.

You can bind IP addresses to an SSL VPN user as follows:

- Bind a list of IP addresses to the user. When the user accesses the SSL VPN gateway in IP access mode, the SSL VPN gateway assigns a bound IP address to the user.
- Enable the SSL VPN gateway to automatically bind the specified number of free addresses in the IP access address pool to the user.

Restrictions and guidelines

The IP addresses to be bound to an SSL VPN user must meet the following requirements:

- If an IP access address pool is specified for the SSL VPN policy group authorized to the user, the IP addresses must exist in the address pool.
- If no address pool is specified for the SSL VPN policy group, the IP addresses must exist in the address pool specified for the SSL VPN context of the user.

You can bind the same IP address to different SSL VPN users only when the SSL VPN contexts of the users are associated with different networks (public network or VPN).

Procedure

1. Enter system view.
   `system-view`
2. Enter SSL VPN context view.
   `sslvpn context` *`context-name`*
3. Create an SSL VPN user and enter SSL VPN user view.
   `user` *`username`*
4. Bind IP addresses to the SSL VPN user.
   `ip-tunnel bind address` { *`ip-address-list`* | **`auto-allocate`** *`number`* }
   By default, an SSL VPN user does not have bound IP addresses.

Configuring SSL VPN access for mobile clients

SSL VPN access for mobile clients tasks at a glance

To configure SSL VPN access for mobile clients, perform the following tasks:

1. Specifying an EMO server for mobile clients
2. (Optional.) Specifying a message server for mobile clients

Specifying an EMO server for mobile clients

About this task

An EMO server provides services for mobile clients. After you specify an EMO server for mobile clients, the SSL VPN gateway issues the EMO server information to the clients. The clients can access available service resources through the EMO server.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Specify an EMO server for mobile clients.

   emo-server address { *host-name* | *ipv4-address* } **port** *port-number*

   By default, no EMO server is specified for mobile clients.

Specifying a message server for mobile clients

About this task

A message server provides services for mobile clients. After you specify a message server for mobile clients, the SSL VPN gateway issues the message server information to the clients. The clients can access the message server.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Specify a message server for mobile clients.

   message-server address { *host-name* | *ipv4-address* } **port** *port-number*

   By default, no message server is specified for mobile clients.

Configuring shortcuts

About this task

To provide quick access to resources on internal servers, configure shortcuts for these resources. A shortcut provides the access link to a protected resource on the SSL VPN Web page. Users can click a shortcut name on the SSL VPN Web page to access the associated resource.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a shortcut and enter its view.

   shortcut *shortcut-name*

   By default, no shortcuts exist.

4. (Optional.) Configure a description for the shortcut.

   description *text*

   By default, no description is configured for a shortcut.

5. Configure a resource link for the shortcut.

   execution *script*

   By default, no resource link is configured for a shortcut.

6. Return to SSL VPN context view.

   quit

7. Create a shortcut list and enter its view.

   shortcut-list *list-name*

8. Assign the shortcut to the shortcut list.

   resources shortcut *shortcut-name*

   By default, a shortcut list does not contain shortcuts.

9. Return to SSL VPN context view.

   quit

10. Enter SSL VPN policy group view.

    policy-group *group-name*

11. Assign the shortcut list to the SSL VPN policy group.

    resources shortcut-list *list-name*

    By default, an SSL VPN policy group does not contain a shortcut list.

Configuring redirect resources

About this task

By default, a user enters the SSL VPN webpage after logging in to the SSL VPN gateway. To provide quick access to the specified Web resource on internal servers, configure the resource as a redirect resource. Users will directly enter the specified redirect resource after a short stay on the SSL VPN Web page.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enter SSL VPN policy group view.

   policy-group *group-name*

4. Configure the Web resource for SSL VPN users to access after login.

   redirect-resource { **shortcut** | **url-item** } *resource-name*

   By default, after logging in to the SSL VPN gateway, a user directly enters the SSL VPN webpage, and no redirection is performed.

Configuring HTTP redirection

About this task

An SSL VPN gateway communicates with users through HTTPS. To allow HTTP to access the SSL VPN gateway, you must configure HTTP redirection.

HTTP redirection enables an SSL VPN gateway to perform the following operations:

1. Listen to an HTTP port.
2. Redirect HTTP requests with the port number to the port used by HTTPS.
3. Send redirection packets to clients.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN gateway view.

   sslvpn gateway *gateway-name*

3. Enable HTTP redirection.

   http-redirect [**port** *port-number*]

   By default, HTTP redirection is disabled. An SSL VPN gateway does not process HTTP traffic.

Configuring the default policy group for an SSL VPN context

About this task

If the AAA server or custom authentication server does not authorize a policy group to a user after the user logs in, the SSL VPN gateway authorizes the default policy group to the user. If no default policy group is configured, the SSL VPN gateway denies all access requests from the user.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create an SSL VPN policy group and enter SSL VPN policy group view.

   policy-group *group-name*

4. Configure accessible resources in the policy group:

   - Configure Web access resources.

     resources url-list *url-list-name*

     By default, no Web access resources are configured in a policy group.

   - Configure TCP access resources.

     resources port-forward *port-forward-name*

     By default, no TCP access resources are configured in a policy group.

   - Configure IP access resources.

     ip-tunnel access-route { *ip-address* { *mask-length* | *mask* } | **force-all** | **ip-route-list** *list-name* }

     By default, no IP access resources are configured in a policy group.

5. (Optional.) Specify the ACLs for Web access filtering:

   - Specify an advanced ACL for Web access filtering.

     filter web-access [**ipv6**] **acl** *advanced-acl-number*

   - Specify a URI ACL for Web access filtering.

     filter web-access uri-acl *uri-acl-name*

   By default, users can access only the Web resources authorized to them through the URL list.

6. (Optional.) Specify the ACLs for TCP access filtering:

   - Specify an advanced ACL for TCP access filtering.

     filter tcp-access [**ipv6**] **acl** *advanced-acl-number*

- Specify a URI ACL for TCP access filtering.

  filter tcp-access uri-acl *uri-acl-name*

  By default, users can access only the TCP resources authorized to them through the TCP port forwarding list.

7. (Optional.) Specify the ACLs for IP access filtering:
   - Specify an advanced ACL for IP access filtering.

     filter ip-tunnel [**ipv6**] **acl** *advanced-acl-number*
   - Specify a URI ACL for IP access filtering.

     filter ip-tunnel uri-acl *uri-acl-name*

   By default, an SSL VPN gateway permits all IP access requests.
8. Return to SSL VPN context view.

   quit
9. Specify the policy group as the default policy group for the SSL VPN context.

   default-policy-group *group-name*

   By default, no default policy group is specified for an SSL VPN context.

Configuring VRF-aware SSL VPN

Associating an SSL VPN context with a VPN instance

About this task

You can associate different SSL VPN contexts with different VPN instances on the SSL VPN gateway. Users in an SSL VPN context can access only the resources in the VPN instance associated with the SSL VPN context. VRF-aware SSL VPN contexts also allow server addresses to overlap.

Prerequisites

Before you configure this feature, complete the following tasks:

- Create the VPN instance.
- Associate the SSL VPN gateway's interface connected to the internal server with the VPN instance.
- (Required for IP access.) Associate the SSL VPN AC interface specified by the **ip-tunnel interface** command with the VPN instance.

For more information about VPN instances, see *VPN Instance Configuration Guide.*

Procedure

1. Enter system view.

   system-view
2. Enter SSL VPN context view.

   sslvpn context *context-name*
3. Associate the SSL VPN context with a VPN instance.

   vpn-instance *vpn-instance-name*

   By default, an SSL VPN context is associated with the public network.

Specifying a VPN instance for an SSL VPN gateway

About this task

After you specify a VPN instance for an SSL VPN gateway, only users in the specified VPN can access the SSL VPN gateway. The VRF-aware SSL VPN gateway prevents the internal server resources from leaking into the public network or other VPNs.

Prerequisites

Before you configure this feature, complete the following tasks:

- Create the VPN instance.
- Associate the VPN instance with the SSL VPN gateway's interface connected to the user.
- Bind the SSL VPN AC interface to

For more information

Procedure

1. Enter system view.
   system-view
2. Enter SSL VPN gateway view.
   sslvpn gateway *gateway-name*
3. Specify a VPN instance for the gateway.
   vpn-instance *vpn-instance-name*

   By default, an SSL VPN gateway belongs to the public network.

Configuring online SSL VPN user control

About this task

Perform this task to configure the SSL VPN user login control features, such as the force logout feature, the maximum number of concurrent logins for each account, and the maximum number of connections allowed per session.

Procedure

1. Enter system view.
   system-view
2. Enter SSL VPN context view.
   sslvpn context *context-name*
3. Force online users to log out.
   force-logout [**all** | **session** *session-id* | **user** *user-name*]
4. Set the maximum number of concurrent logins for each account.
   max-onlines *number*

   By default, the maximum number of concurrent logins for each account is 32.
5. Enable the force logout feature.
   force-logout max-onlines enable

   By default, the force logout feature is disabled. A user cannot log in if the number of logins using the account reaches the maximum.

   When a login is attempted but logins using the account reach the maximum, this feature logs out the user with the longest idle time to allow the new login.

6. Set the maximum number of connections allowed per session.

   session-connections *number*

   By default, a maximum of 64 connections are allowed per session.

   If the number of connections in a session has reached the maximum, new connection requests for the session will be rejected with a **503 Service Unavailable** message.

Configuring SSL VPN session rate limit

About this task

Perform this task to set a rate limit for SSL VPN session upstream and downstream traffic, respectively. If the SSL VPN session upstream or downstream traffic exceeds the rate limit, subsequent upstream or downstream traffic will be discarded.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Set a rate limit for SSL VPN session upstream or downstream traffic.

   rate-limit { **downstream** | **upstream** } *value*

   By default, no rate limit is set for SSL VPN session upstream or downstream traffic.

Configuring SSL VPN cracking prevention

About this task

This feature reduces the risk of brute-force cracking of user login information by limiting the number of login attempts from the same IP address.

If the number of consecutive login failures of the same IP address reaches the specified number, the IP address will be frozen for the specified period of time. During the freeze period, the IP address is prohibited from logging in to the SSL VPN context. When the freeze period expires, the frozen IP address will be unfrozen automatically. To unfreeze the frozen IP address immediately, execute the **prevent-cracking unfreeze-ip** command.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Enable IP address freezing for cracking prevention.

   prevent-cracking freeze-ip enable

   By default, IP address freezing for cracking prevention is disabled.

4. (Optional.) Specify the maximum number of consecutive login failures allowed for an IP address and the period of time to freeze an IP address for cracking prevention.

   prevent-cracking freeze-ip login-failures *login-failures* **freeze-time** *freeze-time*

   By default, the maximum number of consecutive login failures allowed for an IP address is 64, and the period of time to freeze an IP address is 30 seconds.

5. Enable code verification for cracking prevention.

   prevent-cracking verify-code enable

   By default, code verification for cracking prevention is disabled.

6. (Optional.) Specify the maximum number of consecutive login failures allowed for an IP address before performing code verification to prevent cracking.

   prevent-cracking verify-code login-failures *login-failures*

   By default, a maximum of five consecutive login failures are allowed for an IP address before performing code verification.

7. (Optional.) Unfreeze frozen IP addresses.

   prevent-cracking unfreeze-ip { **all** | { **ipv4** | **ipv6** } *ip-address* }

Configuring SSL VPN SSO login

About configuring SSL VPN SSO login

SSO allows a user to use one set of login credentials (such as username and password) to access multiple trusted systems. With SSO, SSL VPN Web access users can gain access to internal servers without entering the login credentials for the internal servers. The device supports the following methods for SSO login:

- **Auto-build method (automatically build login requests)**

  Use a packet capture tool to obtain internal server login requests, and then configure SSO login settings based on the login requests to automatically build login requests to the internal servers. SSO login settings include the HTTP request method, login request encoding method, login parameters, and login data encryption file.

- **Basic authentication method**

  Basic authentication is a simple HTTP authentication scheme, which requires a Web client to enter a username and password to access the server. The server authenticates the client based on the username and password.

  To implement SSO in the basic authentication method, the SSL VPN gateway acts as a Web client and automatically enters a username and password to perform HTTP basic authentication. The entered username and password can be SSL VPN username and password or custom username and password.

  The basic authentication SSO method is applicable only for logging in to the internal servers that support basic authentication.

Restrictions and guidelines

For the auto-build SSO method, the following requirements must be met:

- SSO login is available only for SSL VPN Web access users.
- If a user group name is specified as the SSO login parameter, only remote users are supported.
- SSO login is available only for accessing resources by clicking the URL links on the SSL VPN Web interface. SSO does not work if you access the resources by entering the URLs in a browser address bar or a URL input box.
- SSO login is not available for Web resources that require graphic verification codes.
- SSO login is not available for Web resources that require two-factor authentication or script invocation.

Configuring SSO login in auto-build method

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a URL item and enter its view.

   url-item *name*

4. Specify the resource URL in the URL item.

   url *url*

   By default, no resource URL is specified in a URL item.

   If you do not specify a protocol type in the resource URL, the default protocol (HTTP) is used.

5. Enable Web access SOO and specify the auto-build method.

   sso method auto-build

   By default, Web access SSO login is disabled.

6. Specify the HTTP request method for sending SSO login requests.

   sso auto-build request-method { **get** | **post** }

   By default, the GET request method is used for sending SSO login requests.

7. Specify an encoding method for SSO login requests.

   sso auto-build code { **gb18030** | **utf-8** }

   By default, UTF-8 encoding is used for SSO login requests.

8. Configure a login parameter for automatic building of SSO login requests.

   sso auto-build login-parameter { **cert-fingerprint** | **cert-serial** | **cert-title** | **custom-password** | **custom-username** | **login-name** | **login-password** | **mobile-num** | **user-group** } **name** *parameter-name* [**encrypt**]

   By default, no login parameter is configured for automatic building of SSO login requests.

9. Configure a custom login parameter for automatic building of SSO login requests.

   sso auto-build custom-login-parameter name *parameter-name* **value** *value* [**encrypt**]

   By default, no custom parameter is configured for automatic building of SSO login requests.

10. Specify an encryption file to encrypt the values of parameters in SSO login requests.

    sso auto-build encrypt-file *filename*

    By default, no encryption file is specified.

Configuring SSO login through basic authentication

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Create a URL item and enter its view.

   url-item *name*

4. Specify the resource URL in the URL item.

   url *url*

By default, no resource URL is specified in a URL item.

If you do not specify a protocol type in the resource URL, the default protocol (HTTP) is used.

5. Enable Web access SSO and specify the SSO method as basic authentication.

   ```
   sso method basic
   ```

   By default, Web access SSO login is disabled.

6. (Optional.) Enable using a custom username and password for SSO login through basic authentication.

   ```
   sso basic custom-username-password enable
   ```

   By default, SSL VPN login username and password are used for SSO login through basic authentication.

Configuring WeChat Work authentication

About WeChat Work authentication

WeChat Work (or WeCom) authentication allows the device to obtain user information in a company from WeChat Work and uses the user information for user authentication and authorization. This feature is transparent to users in the company.

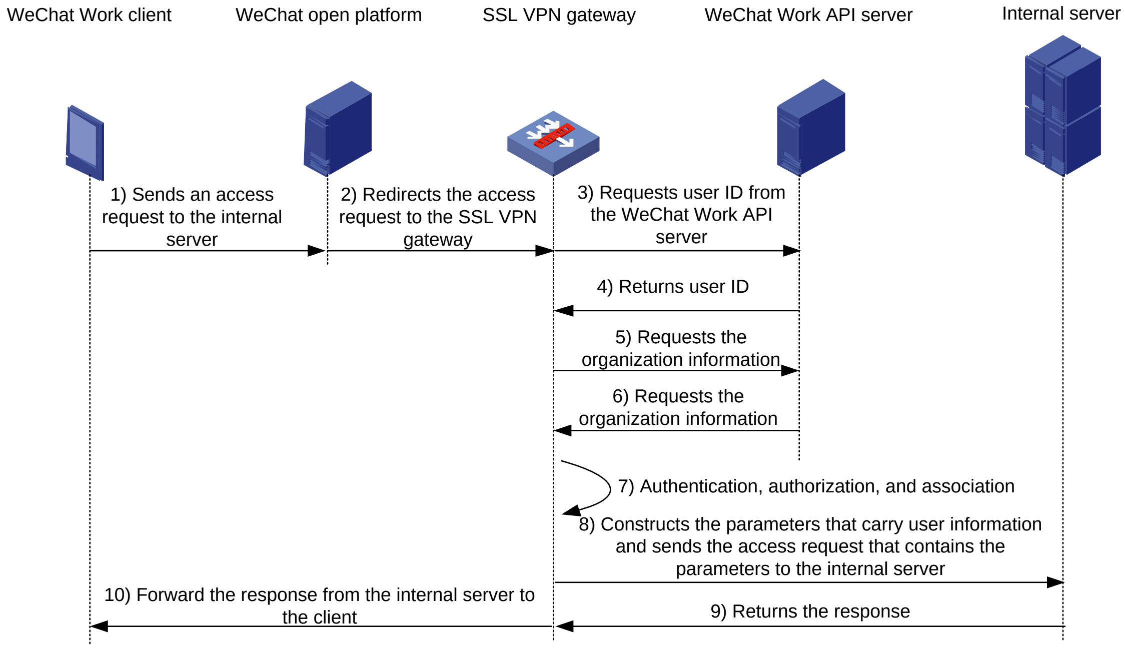
As shown in Figure 11, WeChat Work authentication operates as follows:

1. A user in a company uses the WeChat Work client to access an internal resource. The client sends the resource access request to the WeChat open platform.
2. The WeChat open platform redirects the request to the SSL VPN gateway for the gateway to protect the internal resource.

   Make sure the redirect link has been configured on the WeChat open platform.
3. On receiving the packet redirected from the WeChat Work server, the SSL VPN gateway sends a request to the WeChat Work API server to obtain the user ID.
4. The WeChat Work API server returns the user ID.
5. The SSL VPN gateway uses the user ID to further obtain the organization information of the user from the WeChat Work API server.

   The organization information corresponds to the authorization policy group name configured on the SSL VPN gateway.
6. The WeChat Work API server returns the organization information.
7. Based on the obtained user information, the SSL VPN gateway performs authentication for the user and authorizes the user to access the internal resource.
8. The SSL VPN gateway constructs a login request with parameters that carry the user information, and sends the request to the internal server.
9. The internal server returns the response to the SSL VPN gateway.
10. The SSL VPN gateway forwards the response to the WeChat Work client. The user then can access the internal resource through the client.

Figure 11 WeChat Work authentication mechanism


WeChat Work client
WeChat open platform
SSL VPN gateway
WeChat Work API server
Internal server
1) Sends an access request to the internal server
2) Redirects the access request to the SSL VPN gateway
3) Requests user ID from the WeChat Work API server
4) Returns user ID
5) Requests the organization information
6) Requests the organization information
7) Authentication, authorization, and association
8) Constructs the parameters that carry user information and sends the access request that contains the parameters to the internal server
9) Returns the response
10) Forward the response from the internal server to the client

Restrictions and guidelines

The self-signed certificate on the device does not support WeChat Work authentication. To use WeChat Work authentication, install a trusted SSL certificate first.

To enable WeChat Work authentication for an SSL VPN context, you must associate the SSL VPN context to an SSL VPN gateway exclusively.

Prerequisites

Before configuring WeChat Work authentication, you must configure the app homepage redirect link and the trusted domain name of the SSL VPN gateway for each app on the WeChat Work management platform.

Configuring the app homepage redirect link for an app

1. Enter **https://work.weixin.qq.com** in the browser.
2. Use the WeChat Work client to scan the QR code to log in to the WeChat Work management platform.
3. On the WeChat Work management platform, click **App Management** and select an app.
4. In the **Workplace App Management** area, click **Enabled** to configure the app homepage redirect link in the format of https://open.weixin.qq.com/connect/oauth2/authorize?appid=CORPID&redirect_uri=https://gateway.com:port/_proxywx/http/80/www.resources.com/?ctx=contextName&response_type=code&scope=snsapi_base&agentid=AGENTID&connect_redirect=1#wechat_redirect.
   - *CORPID*—Company ID. To view the company ID, go to **My Company** > **Company Information**.
   - *gateway.com:port*—Domain name and port number of the SSL VPN gateway.
   - www.resources.com—Domain name of the internal resource.
   - *contextName*—SSL VPN context name.

- *AGENTID*—App ID. To view the app ID of an app, select the app on the **App Management** page, and view the **Agentid** field.

You must encode https://*gateway.com:port*/_proxywx/http/80/*www.resources.com*/?ctx=*contextName* to the URL encoding format.

Configuring the trusted domain name of the SSL VPN gateway for an app

1. Enter **https://work.weixin.qq.com** in the browser.
2. Use the WeChat Work client to scan the QR code to log in to the WeChat Work management platform.
3. On the WeChat Work management platform, click **App Management** and select an app.
4. In the **Web Authorization and JS-SDK** area, click **Apply for domain name verification**. In the window that opens, enter the domain name and port number of the SSL VPN gateway in the format of *gateway.com:port* in the **Trustable Domain Names** field.
5. Click **Domain name to be verified** and download the verification file as instructed. Then, upload the verification file on the SSL VPN Web interface.
6. Select **The domain name ownership verification file has been uploaded.**, and then click **Confirm**.

Procedure

1. Enter system view.

   `system-view`
2. Enter SSL VPN context view.

   `sslvpn context` *`context-name`*
3. Enable WeChat Work authentication.

   `wechat-work-authentication enable`

   By default, WeChat Work authentication is disabled.
4. Specify the URL of the WeChat Work API server.

   `wechat-work-authentication url` *`url`*

   By default, no WeChat Work API server URL is configured.
5. Specify the WeChat Work authentication timeout.

   `wechat-work-authentication timeout` *`seconds`*

   By default, the WeChat Work authentication timeout is 15 seconds.
6. Specify the company ID for WeChat Work authentication.

   `wechat-work-authentication corp-id` *`corp-id`*

   By default, no company ID is specified for WeChat Work authentication.
7. Specify the app secret key for WeChat Work authentication.

   `wechat-work-authentication app-secret` *`app-secret`*

   By default, no app secret key is specified for WeChat Work authentication.
8. Specify the user ID field name for the SSL VPN gateway to access the internal server.

   `wechat-work-authentication userid-field` *`userid-field`*

   By default, no user ID field name is configured for the SSL VPN gateway to access the internal server.
9. Specify the name of the authorization policy group field.

   `wechat-work-authentication authorize-field` *`authorize-field`*

By default, no authorization policy group field name is specified for WeChat Work authentication.

10. Specify the WeChat open platform URL.

    wechat-work-authentication open-platform-url { **pre-defined** | **user-defined** *user-defined-url* }

    By default, no WeChat open platform URL is specified.

Customizing SSL VPN webpages

Restrictions and guidelines

If a user-defined webpage template is specified in an SSL VPN context, all other webpage customization settings are invalid for the SSL VPN context.

Customizing SSL VPN webpage elements

About this task

You can customize the following elements on the SSL VPN webpage:

- Login message.
- Password input box displaying.
- Title.
- Logo.
- Notification message on the SSL VPN gateway login page and resource page.
- Files for users to download on the SSL VPN resource page.
- Password complexity description.
- Server reply message rewriting.

Procedure

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Configure a login message.

   login-message { **chinese** *chinese-message* | **english** *english-message* }

   By default, the login message is **Welcome to SSL VPN**.

4. Hide the password input box on the SSL VPN Web login page.

   password-box hide

   By default, the password input box is displayed on the SSL VPN Web login page.

5. Configure a title.

   title { **chinese** *chinese-title* | **english** *english-title* }

   By default, the title is **SSL VPN**.

6. Specify a logo.

   logo { **file** *file-name* | **none** }

7. Configure the notification message to be displayed on the SSL VPN gateway login page or resource page.

notify-message { **login-page** | **resource-page** } { **chinese** *chinese-message* | **english** *english-message* }

By default, no notification message is configured.

8. Specify a file for users to download on the SSL VPN gateway resource page.

   resources-file { **chinese** *chinese-filename* | **english** *english-filename* }

   By default, no file is provided for users to download.

9. Configure the password complexity message to be displayed on the SSL VPN password modification page.

   password-complexity-message { **chinese** *chinese-message* | **english** *english-message* }

   By default, no password complexity message is configured.

10. Rewrite a server reply message.

    rewrite server-response-message *server-response-message* { **chinese** *chinese-message* | **english** *english-message* }

    By default, no server reply message is rewritten.

Specifying an SSL VPN webpage template

About this task

This task allows you to customize SSL VPN webpages by specifying an SSL VPN webpage template. An SSL VPN webpage template defines the style of the SSL VPN gateway login page and resource page.

You can specify a webpage template in system view and in SSL VPN context view.

- The webpage template set in system view is the global SSL VPN webpage template, which is applicable to all SSL VPN contexts.
- The webpage template set in SSL VPN context view is applicable only to the current SSL VPN context.

Prerequisites

Upload the user-defined webpage templates to the file system of the device from the webpage.

Restrictions and guidelines for SSL VPN webpage customization

The SSL VPN webpage template specified in SSL VPN context view takes precedence over that in system view.

Specifying an SSL VPN webpage template in system view

1. Enter system view.

   system-view

2. Specify the global SSL VPN webpage template.

   sslvpn webpage-customize *template-name*

   By default, no global SSL VPN webpage template is specified. SSL VPN uses the system default SSL VPN webpages.

Specifying an SSL VPN webpage template in an SSL VPN context

1. Enter system view.

   system-view

2. Enter SSL VPN context view.

   sslvpn context *context-name*

3. Specify an SSL VPN webpage template.

   webpage-customize *template-name*

   By default, no SSL VPN webpage template is specified for an SSL VPN context. An SSL VPN context uses the global SSL VPN webpage template.

Enabling SSL VPN logging

About this task

Logs generated by SSL VPN logging are sent to the information center of the device. For the logs to be output correctly, you must also configure the information center on the device. For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable the SSL VPN global logging feature.

   sslvpn log enable

   By default, the SSL VPN global logging feature is disabled.

3. Enter SSL VPN context view.

   sslvpn context *context-name*

4. Enable logging for user login and logoff events.

   log user-login enable

   By default, logging for user login and logoff events is disabled.

5. Enable logging for resource accesses of users.

   log resource-access enable [**brief** | **filtering**] *

   By default, resource access logging is disabled.

6. Enable logging for IP access connection close events.

   ip-tunnel log connection-close

   By default, logging for IP access connection close events is disabled.

7. Enable logging for IP access packet drop events.

   ip-tunnel log packet-drop

   By default, logging for IP access packet drop events is disabled.

8. Enable logging for IP address allocations and releases for the VNIC of the IP access client.

   ip-tunnel log address-alloc-release

   By default, logging is disabled for IP address allocations and releases for the VNIC of the IP access client.

Display and maintenance commands for SSL VPN

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display SSL VPN AC interface information.	**display interface sslvpn-ac**

	[*interface-number*] [**brief** [**description** | **down**]]
Display SSL VPN context information.	**display sslvpn context** [**brief** | **name** *context-name*]
Display SSL VPN gateway information.	**display sslvpn gateway** [**brief** | **name** *gateway-name*]
Display packet statistics for IP access users.	**display sslvpn ip-tunnel statistics** [**context** *context-name*] [**user** *user-name*]
Display SSL VPN policy group information.	**display sslvpn policy-group** *group-name* [**context** *context-name*]
Display TCP port forwarding connection information.	**display sslvpn port-forward connection** [**context** *context-name*] [**slot** *slot-number*]
Display information about IP addresses frozen for cracking prevention.	**display sslvpn prevent-cracking frozen-ip** { **statistics** | **table** } [**context** *context-name*]
Display SSL VPN session information.	**display sslvpn session** [**context** *context-name*] [**user** *user-name* | **verbose**]
Display SSL VPN webpage template information.	**display sslvpn webpage-customize template**
Clear SSL VPN AC interface statistics.	**reset counters interface** [**sslvpn-ac** [*interface-number*]]
Clear packet statistics for IP access users.	**reset sslvpn ip-tunnel statistics** [**context** *context-name* [**session** *session-id*]]

SSL VPN configuration examples

Example: Configuring Web access

Network configuration

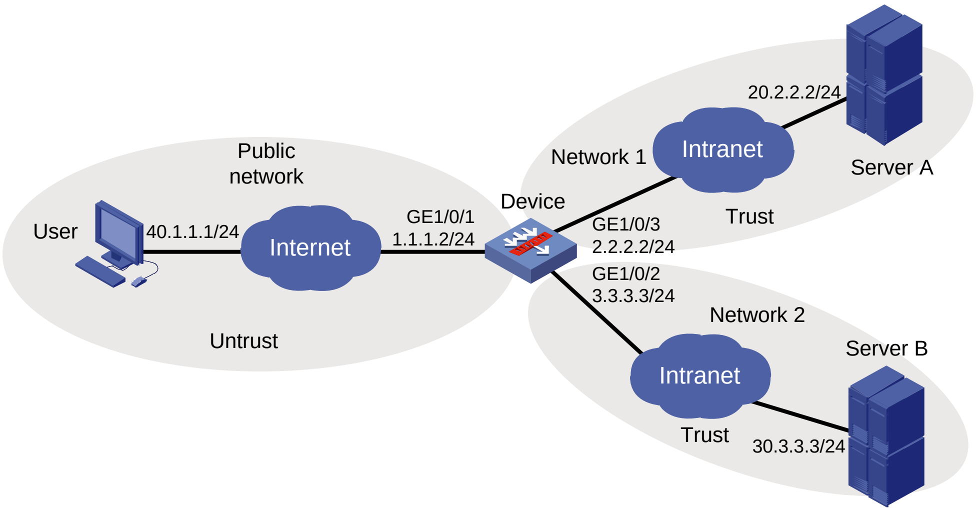
As shown in Figure 12, the device acts as the SSL VPN gateway that connects the public network and private networks Network 1 and Network 2. Server A and Server B are internal Web servers. Server A uses HTTP over port 80. Server B uses HTTPS over port 443.

The device uses a CA-signed SSL server certificate. If no SSL server policy is applied to the device, the device uses a self-signed SSL server certificate.

Configure SSL VPN Web access on the device to allow the user to access Server A in Network 1 and Server B in Network 2.

Configure the device to perform local authentication and authorization for the user.

Figure 12 Network diagram



Procedure

1. Obtain CA certificate file **ca.cer** and local certificate file **server.pfx** for the device. (Details not shown.)
2.
3. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

4. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3 to Server A, 3.3.3.4 to Server B, and 1.1.1.3 to the user.

```
[Device] ip route-static 20.2.2.2 24 2.2.2.3
[Device] ip route-static 30.3.3.3 24 3.3.3.4
[Device] ip route-static 40.1.1.1 24 1.1.1.3
```

5. Add interfaces to security zones.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] import interface gigabitethernet 1/0/3
[Device-security-zone-Trust] quit
```

6. Configure rules in a security policy to permit the traffic between the **Untrust** and **Local** security zones for the user to access the SSL VPN gateway:

   # Configure a rule named **sslvpnlocalout1** to permit the packets from the device to the user.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name sslvpnlocalout1
[Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
[Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
```

```
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
[Device-security-policy-ip-1-sslvpnlocalout1] action pass
[Device-security-policy-ip-1-sslvpnlocalout1] quit
```

# Configure a rule named **sslvpnlocalin1** to permit the packets from the user to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin1
[Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
[Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
[Device-security-policy-ip-2-sslvpnlocalin1] action pass
[Device-security-policy-ip-2-sslvpnlocalin1] quit
```

# Configure a rule named **sslvpnlocalout2** to permit the packets from the device to Server A or Server B.

```
[Device-security-policy-ip] rule name sslvpnlocalout2
[Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 3.3.3.3
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 30.3.3.3
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

# Configure a rule named **sslvpnlocalin2** to permit the packets from Server A and Server B to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 30.3.3.3
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 3.3.3.3
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
[Device-security-policy-ip] quit
```

7. Configure a PKI domain named **sslvpn** and certificate-related parameters.

```
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

8. Create an SSL server policy named **ssl** and specify PKI domain **sslvpn** for the policy.

```
[Device] ssl server-policy ssl
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] quit
```

9. Configure the SSL VPN gateway for user access. Configure the IP address for SSL VPN gateway **gw** as 1.1.1.2 and port number as 2000, and then apply server policy **ssl** to the gateway.

```
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 2000
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

10. Configure SSL VPN contexts to provide Web access service:

    # Create SSL VPN context **ctx1**, specify gateway **gw** and domain **domain1** for the context, and then associate the context with VPN instance **VPN1**.

    ```
    [Device] sslvpn context ctx1
    [Device-sslvpn-context-ctx1] gateway gw domain domain1
    [Device-sslvpn-context-ctx1] vpn-instance VPN1
    [Device-sslvpn-context-ctx1] url-item urlitem
    [Device-sslvpn-context-ctx1-url-item-urlitem] url http://20.2.2.2
    [Device-sslvpn-context-ctx1-url-item-urlitem] quit
    [Device-sslvpn-context-ctx1] url-list urllist
    [Device-sslvpn-context-ctx1-url-list-urllist] heading web
    [Device-sslvpn-context-ctx1-url-list-urllist] resources url-item urlitem
    [Device-sslvpn-context-ctx1-url-list-urllist] quit
    [Device-sslvpn-context-ctx1] policy-group pgroup
    [Device-sslvpn-context-ctx1-policy-group-pgroup] resources url-list urllist
    [Device-sslvpn-context-ctx1-policy-group-pgroup] quit
    [Device-sslvpn-context-ctx1] default-policy-group pgroup
    [Device-sslvpn-context-ctx1] service enable
    [Device-sslvpn-context-ctx1] quit
    ```

    # Create SSL VPN context **ctx2**, specify gateway **gw** and domain **domain2** for the context, and then associate the context with VPN instance **VPN2**.

    ```
    [Device] sslvpn context ctx2
    [Device-sslvpn-context-ctx2] gateway gw domain domain2
    [Device-sslvpn-context-ctx2] vpn-instance VPN2
    [Device-sslvpn-context-ctx2] url-item urlitem
    [Device-sslvpn-context-ctx2-url-item-urlitem] url https://30.3.3.3
    [Device-sslvpn-context-ctx2-url-item-urlitem] quit
    [Device-sslvpn-context-ctx2] url-list urllist
    [Device-sslvpn-context-ctx2-url-list-urllist] heading web
    [Device-sslvpn-context-ctx2-url-list-urllist] resources url-item urlitem
    [Device-sslvpn-context-ctx2-url-list-urllist] quit
    [Device-sslvpn-context-ctx2] policy-group pgroup
    [Device-sslvpn-context-ctx2-policy-group-pgroup] resources url-list urllist
    [Device-sslvpn-context-ctx2-policy-group-pgroup] quit
    [Device-sslvpn-context-ctx2] default-policy-group pgroup
    [Device-sslvpn-context-ctx2] service enable
    [Device-sslvpn-context-ctx2] quit
    ```

11. Create a local user named **sslvpn**, set the password to **123456**, service type to **sslvpn**, and user role to **network-operator**. Authorize the user to use policy group **pgroup**.

    ```
    [Device] local-user sslvpn class network
    [Device-luser-network-sslvpn] password simple 123456
    [Device-luser-network-sslvpn] service-type sslvpn
    [Device-luser-network-sslvpn] authorization-attribute user-role network-operator
    [Device-luser-network-sslvpn] authorization-attribute sslvpn-policy-group pgroup
    ```

```
    [Device-luser-network-sslvpn] quit
```

Verifying the configuration

# Verify that SSL VPN gateway **gw** is up on the device.

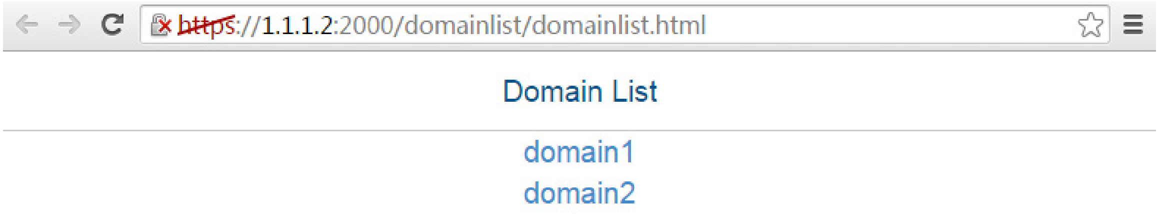
```
[Device] display sslvpn gateway
Gateway name: gw
  Operation state: Up
  IP: 1.1.1.2  Port: 2000
  SSL server policy configured: ssl
  SSL server policy in use: ssl
  Front VPN instance: Not configured
```

# Verify that SSL VPN contexts **ctx1** and **ctx2** are up on the device.

```
[Device] display sslvpn context
Context name: ctx1
  Operation state: Up
  AAA domain: Not specified
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: pgroup
  Associated SSL VPN gateway: gw
    Domain name: domain1
  SSL client policy configured: ssl
  SSL client policy in use: ssl
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled

Context name: ctx2
  Operation state: Up
  AAA domain: Not specified
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: pgroup
  Associated SSL VPN gateway: gw
    Domain name: domain2
  SSL client policy configured: ssl
  SSL client policy in use: ssl
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled
```

# On the user PC, enter **https://1.1.1.2:2000/** in the browser address bar to open the domain list page.

Figure 13 Domain list page



~~https~~://1.1.1.2:2000/domainlist/domainlist.html

Domain List

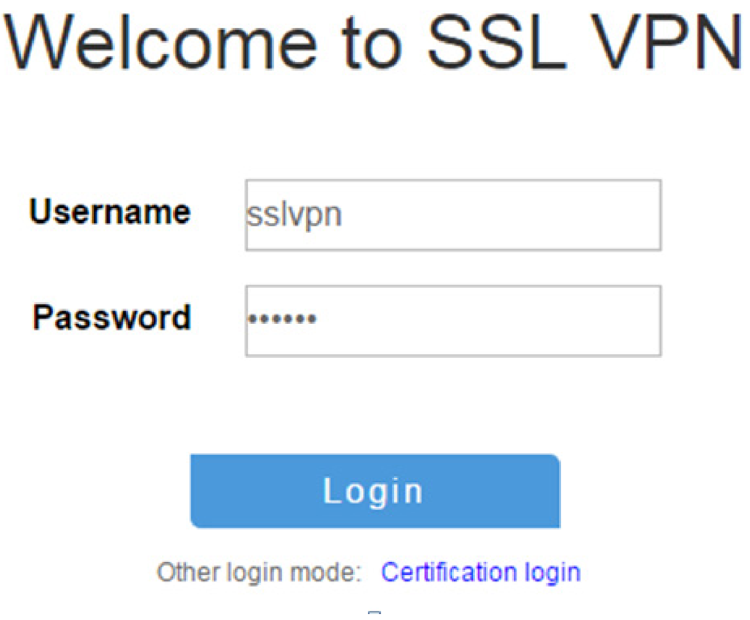
domain1

domain2

# Select **domain1** to enter the login page.

# On the login page, enter username **sslvpn** and password **123456**, and then click **Login**.

Figure 14 Login page



Welcome to SSL VPN

Username: sslvpn

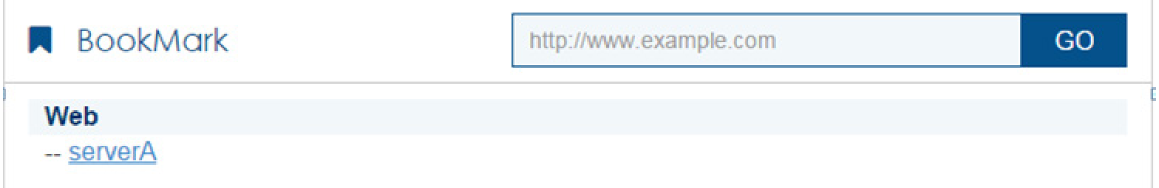
Password: ••••••

# Display SSL VPN session information on the device after the user logged in.

```
[Device] display sslvpn session context ctx1
SSL VPN context: ctx1
Users: 1
Username        Connections  Idle time    Created       User IP
sslvpn          6            0/00:12:05  0/00:04:14    40.1.1.1
```

# On the SSL VPN gateway home page, click the **serverA** link in the **BookMark** area to open the webpage of Server A. The URL **https://1.1.1.2:2000/_proxy2/http/80/20.2.2.2/** is displayed in the browser address bar.

Figure 15 SSL VPN gateway home page



BookMark

http://www.example.com

GO

Web

-- serverA

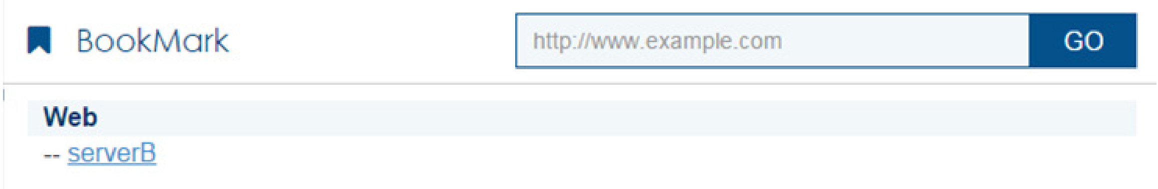
# Log out and restart the browser. Enter **https://1.1.1.2:2000/** to enter the domain list page, and then select **domain2** to enter the login page. On the login page, enter username **sslvpn** and password **123456**, and then click **Login**. (Details not shown.)

# Display SSL VPN session information on the device after the user logged in.

```
[Device] display sslvpn session context ctx2
SSL VPN context: ctx2
Users: 1
Username        Connections  Idle time    Created       User IP
sslvpn          6            0/00:02:05   0/00:01:11    40.1.1.1
```

# On the SSL VPN gateway home page, click the **serverB** link in the **BookMark** area to open the webpage of Server B. The URL **https://1.1.1.2:2000/_proxy2/https/443/30.3.3.3/** is displayed in the browser address bar.

Figure 16 SSL VPN gateway home page



BookMark

http://www.example.com

GO

Web

-- serverB

Example: Configuring TCP access

Network configuration

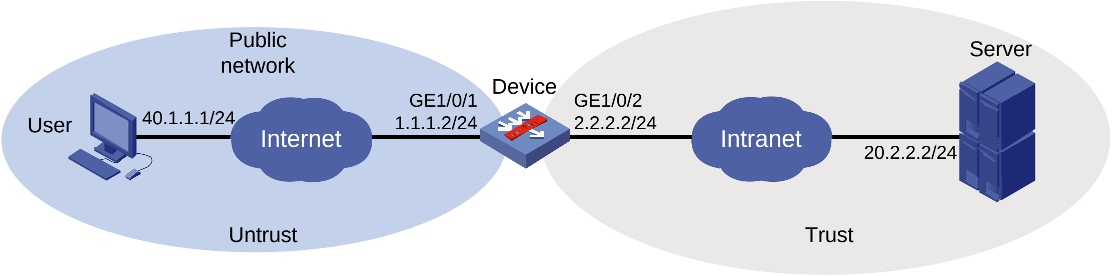
As shown in Figure 17, the device acts as an SSL VPN gateway that connects the public network and private network Network 1.

The device uses a CA-signed SSL server certificate. If no SSL server policy is applied to the device, the device uses a self-signed SSL server certificate.

Configure SSL VPN TCP access on the device to allow the user to access the internal Telnet server in Network 1.

Configure the device to perform local authentication and local authorization for the user.

Figure 17 Network diagram



Prerequisites

Before using the user's PC to access the SSL VPN gateway (the device), make sure a Java running environment is installed on the PC.

Procedure

1. Obtain CA certificate file **ca.cer** and local certificate file **server.pfx** for the device. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3 to the server, and 1.1.1.3 to the user.

   ```
   [Device] ip route-static 20.2.2.2 24 2.2.2.3
   [Device] ip route-static 40.1.1.1 24 1.1.1.3
   ```
4. Add interfaces to security zones.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```
5. Configure rules in a security policy to permit the traffic between the **Untrust** and **Local** security zones for the user to access the SSL VPN gateway:

   # Configure a rule named **sslvpnlocalout1** to permit the packets from the device to the user.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name sslvpnlocalout1
   [Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
   [Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
   [Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
   [Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
   [Device-security-policy-ip-1-sslvpnlocalout1] action pass
   [Device-security-policy-ip-1-sslvpnlocalout1] quit
   ```

   # Configure a rule named **sslvpnlocalin1** to permit the packets from the user to the device.

   ```
   [Device-security-policy-ip] rule name sslvpnlocalin1
   [Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
   [Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
   [Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
   [Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
   [Device-security-policy-ip-2-sslvpnlocalin1] action pass
   [Device-security-policy-ip-2-sslvpnlocalin1] quit
   ```

   # Configure a rule named **sslvpnlocalout2** to permit the packets from the device to the server.

   ```
   [Device-security-policy-ip] rule name sslvpnlocalout2
   [Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
   ```

```
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

# Configure a rule named **sslvpnlocalin2** to permit the packets from the server to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
[Device-security-policy-ip] quit
```

6. Configure a PKI domain named **sslvpn** and certificate-related parameters.

```
<Device> system-view
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

7. Create an SSL server policy named **ssl** and specify PKI domain **sslvpn** for the policy.

```
[Device] ssl server-policy ssl
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] quit
```

8. Configure the SSL VPN gateway for user access. Configure the IP address for SSL VPN gateway **gw** as 1.1.1.2 and port number as 2000, and then apply server policy **ssl** to the gateway.

```
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 2000
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

9. Create an SSL VPN context named **ctx**, specify gateway **gw** for the context, and then associate the context with VPN instance **VPN1**.

```
[Device] sslvpn context ctx
[Device-sslvpn-context-ctx] gateway gw
[Device-sslvpn-context-ctx] vpn-instance VPN1
[Device-sslvpn-context-ctx] port-forward-item pfitem1
[Device-sslvpn-context-ctx-port-forward-item-pfitem1] local-port 2323 local-name
127.0.0.1 remote-server 20.2.2.2 remote-port 23 description telnet
[Device-sslvpn-context-ctx-port-forward-item-pfitem1] quit
[Device-sslvpn-context-ctx] port-forward plist
[Device-sslvpn-context-ctx-port-forward-plist] resources port-forward-item pfitem1
[Device-sslvpn-context-ctx-port-forward-plist] quit
[Device-sslvpn-context-ctx] policy-group pgroup
[Device-sslvpn-context-ctx-policy-group-pgroup] resources port-forward plist
[Device-sslvpn-context-ctx-policy-group-pgroup] quit
```

```
[Device-sslvpn-context-ctx] service enable
[Device-sslvpn-context-ctx] quit
```

10. Create a local user named **sslvpn**, set the password to **123456**, service type to **sslvpn**, and user role to **network-operator**. Authorize the user to use policy group **pgroup**.

```
[Device] local-user sslvpn class network
[Device-luser-network-sslvpn] password simple 123456
[Device-luser-network-sslvpn] service-type sslvpn
[Device-luser-network-sslvpn] authorization-attribute user-role network-operator
[Device-luser-network-sslvpn] authorization-attribute sslvpn-policy-group pgroup
[Device-luser-network-sslvpn] quit
```

Verifying the configuration

# Verify that SSL VPN gateway **gw** is up on the device.

```
[Device] display sslvpn gateway
Gateway name: gw
  Operation state: Up
  IP: 1.1.1.2  Port: 2000
  SSL server policy configured: ssl
  SSL server policy in use: ssl
  Front VPN instance: Not configured
```

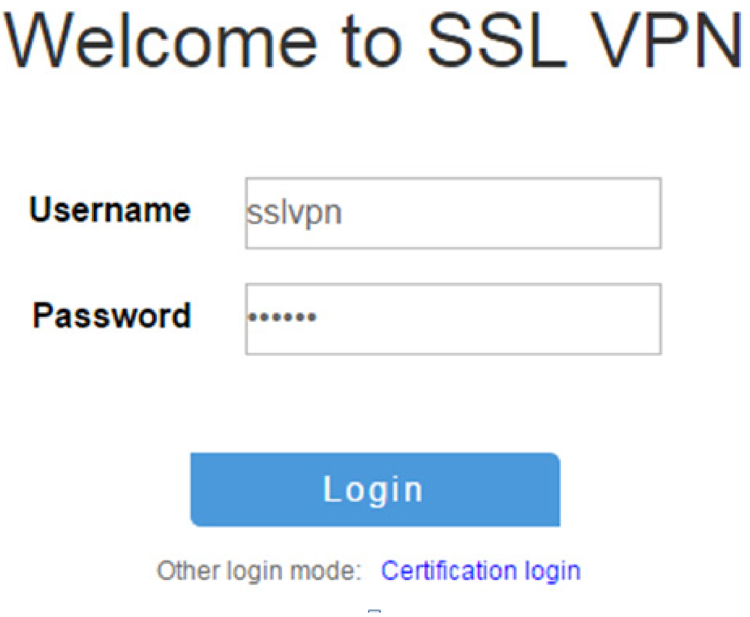
# Verify that SSL VPN context **ctx** is up on the device.

```
[Device] display sslvpn context
Context name: ctx
  Operation state: Up
  AAA domain: Not specified
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: Not configured
  Associated SSL VPN gateway: gw
  SSL client policy configured: ssl
  SSL client policy in use: ssl
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled
```

# On the user PC, enter **https://1.1.1.2:2000/** in the browser address bar to enter login page.

# On the login page, enter username **sslvpn** and password **123456**, and then click **Login**.

Figure 18 Login page



# On the SSL VPN home page that opens, click **Start** to download the TCP client application and start the application.

NOTE:

You cannot start the TCP client application by double-clicking it.

# Telnet to the local address (127.0.0.1) and local port (2323) on the PC. The user can remotely access the server. (Details not shown.)

# Display SSL VPN session information on the device.

```
[Device] display sslvpn session context ctx
SSL VPN context: ctx
Users: 1
Username        Connections  Idle time   Created       User IP
sslvpn          6            0/00:12:05  0/00:04:14    40.1.1.1
```

# Display SSL VPN port forwarding connection information on the device.

```
[Device] display sslvpn port-forward connection
SSL VPN context  : ctx
  Client address : 40.1.1.1
  Client port    : 50788
  Server address : 20.2.2.2
  Server port    : 23
  State          : Connected
```

Example: Configuring IP access

Network configuration

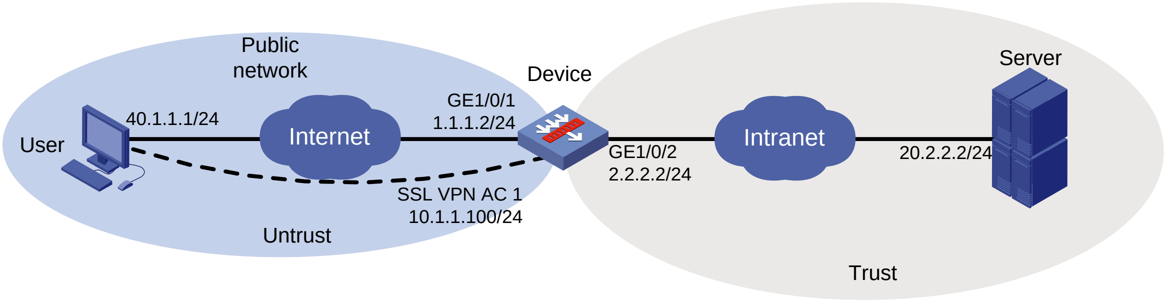
As shown in Figure 19, the device acts as an SSL VPN gateway that connects the public network and the private network.

The device uses a CA-signed SSL server certificate. If no SSL server policy is applied to the device, the device uses a self-signed SSL server certificate.

Configure SSL VPN IP access on the device to allow the user to access the internal server in the private network.

Configure the device to perform local authentication and authorization for the user.

Figure 19 Network diagram



Prerequisites

Before configuring IP access, make sure the server has a route to 10.1.1.0/24.

Procedure

1. Obtain CA certificate file **ca.cer** and local certificate file **server.pfx** for the device. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Create SSL VPN AC interface AC 1 and configure the IP address as 10.1.1.100/24 for the interface.

   ```
   [Device] interface sslvpn-ac 1
   [Device-SSLVPN-AC1] ip address 10.1.1.100 24
   [Device-SSLVPN-AC1] quit
   ```
4. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3 to the server, and 1.1.1.3 to the user.

   ```
   [Device] ip route-static 20.2.2.2 24 2.2.2.3
   [Device] ip route-static 40.1.1.1 24 1.1.1.3
   ```
5. Add interfaces to security zones.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] import interface sslvpn-ac 1
   [Device-security-zone-Untrust] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```

6. Configure rules in a security policy to permit the traffic between security zones for the user to access the SSL VPN gateway and the server:

   # Configure a rule named **sslvpnlocalout1** to permit the packets from the device to the user.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name sslvpnlocalout1
[Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
[Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
[Device-security-policy-ip-1-sslvpnlocalout1] action pass
[Device-security-policy-ip-1-sslvpnlocalout1] quit
```

   # Configure a rule named **sslvpnlocalin1** to permit the packets from the user to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin1
[Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
[Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
[Device-security-policy-ip-2-sslvpnlocalin1] action pass
[Device-security-policy-ip-2-sslvpnlocalin1] quit
```

   # Configure a rule named **sslvpnlocalout2** to permit the packets from the device to the server.

```
[Device-security-policy-ip] rule name sslvpnlocalout2
[Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

   # Configure a rule named **sslvpnlocalin2** to permit the packets from the server to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
```

   # Configure a rule named **untrust-trust** to allow the user to access the server through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-5-untrust-trust] source-zone untrust
[Device-security-policy-ip-5-untrust-trust] destination-zone trust
[Device-security-policy-ip-5-untrust-trust] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-5-untrust-trust] destination-ip-host 20.2.2.2
[Device-security-policy-ip-5-untrust-trust] action pass
[Device-security-policy-ip-5-untrust-trust] quit
```

   # Configure a rule named **trust-untrust** to permit the packets from the server to the user through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-6-trust-untrust] source-zone trust
```

```
[Device-security-policy-ip-6-trust-untrust] destination-zone untrust
[Device-security-policy-ip-6-trust-untrust] source-ip-host 20.2.2.2
[Device-security-policy-ip-6-trust-untrust] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-6-trust-untrust] action pass
[Device-security-policy-ip-6-trust-untrust] quit
[Device-security-policy-ip] quit
```

7. Create ACL 3000. Add a rule to permit the packets sourced from subnet 10.1.1.0/24 and destined for 20.2.2.0/24.

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip source 10.1.1.0 0.0.0.255 destination
20.2.2.0 0.0.0.255
[Device-acl-ipv4-adv-3000] quit
```

8. Configure a PKI domain named **sslvpn** and certificate-related parameters.

```
<Device> system-view
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check  enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

9. Create an SSL server policy named **ssl** and specify PKI domain **sslvpn** for the policy.

```
[Device] ssl server-policy ssl
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] quit
```

10. Configure the SSL VPN gateway for user access. Configure the IP address for SSL VPN gateway **gw** as 1.1.1.2 and port number as 4430, and then apply SSL server policy **ssl** to the gateway.

```
<Device> system-view
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 4430
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

11. Create an IP access address pool named **sslvpnpool** and specify the address range as 10.1.1.1 to 10.1.1.10.

```
[Device] sslvpn ip address-pool sslvpnpool 10.1.1.1 10.1.1.10
```

12. Create SSL VPN context **ctxip**, and then specify gateway **gw** and domain **domainip** for the context.

```
[Device] sslvpn context ctxip
[Device-sslvpn-context-ctxip] gateway gw domain domainip
[Device-sslvpn-context-ctxip] ip-tunnel interface sslvpn-ac 1
[Device-sslvpn-context-ctxip] ip-route-list rtlist
[Device-sslvpn-context-ctxip-route-list-rtlist] include 20.2.2.0 24
[Device-sslvpn-context-ctxip-route-list-rtlist] quit
[Device-sslvpn-context-ctxip] ip-tunnel address-pool sslvpnpool mask 24
[Device-sslvpn-context-ctxip] policy-group resourcegrp
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] ip-tunnel access-route
ip-route-list rtlist
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] filter ip-tunnel acl 3000
```

```
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] quit
[Device-sslvpn-context-ctxip] service enable
[Device-sslvpn-context-ctxip] quit
```

13. Create a local user named **sslvpnuser**, set the password to **123456**, service type to **sslvpn**, and user role to **network-operator**. Authorize the user to use policy group **resourcegrp**.

```
[Device] local-user sslvpnuser class network
[Device-luser-network-sslvpnuser] password simple 123456
[Device-luser-network-sslvpnuser] service-type sslvpn
[Device-luser-network-sslvpnuser] authorization-attribute sslvpn-policy-group
resourcegrp
[Device-luser-network-sslvpnuser] authorization-attribute user-role
network-operator
[Device-luser-network-sslvpnuser] quit
```

Verifying the configuration

# Verify that SSL VPN gateway **gw** is up on the device.

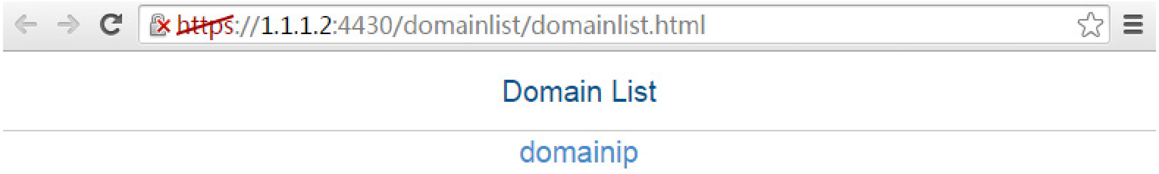
```
[Device] display sslvpn gateway
Gateway name: gw
  Operation state: Up
  IP: 1.1.1.2  Port: 4430
  Front VPN instance: Not configured
```

# Verify that SSL VPN context **ctxip** is up on the device.

```
[Device] display sslvpn context
Context name: ctxip
  Operation state: Up
  AAA domain: Not specified
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: Not configured
  Associated SSL VPN gateway: gw
    Domain name: domainip
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled
```

# On the user PC, enter **https://1.1.1.2:4430/** in the browser address bar to open the domain list page.

Figure 20 Domain list page



~~https~~://1.1.1.2:4430/domainlist/domainlist.html

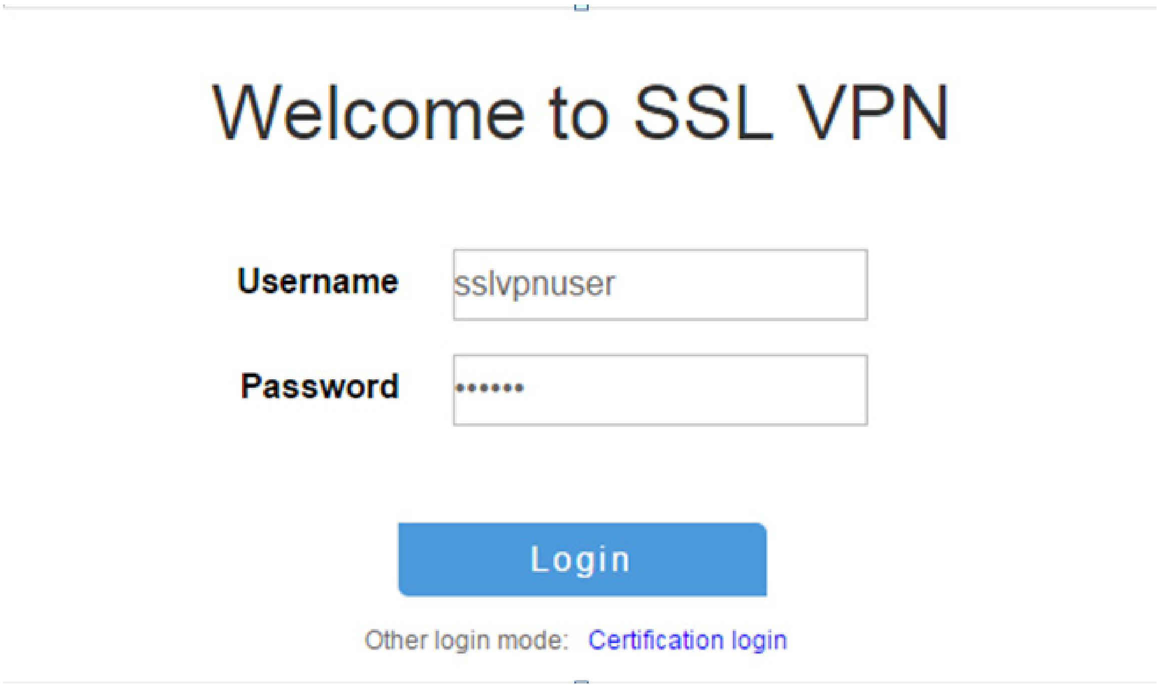
Domain List

domainip

# Select **domainip** to access the login page.

# On the login page, enter username **sslvpnuser** and password **123456**, and then click **Login**.

Figure 21 Login page



Welcome to SSL VPN

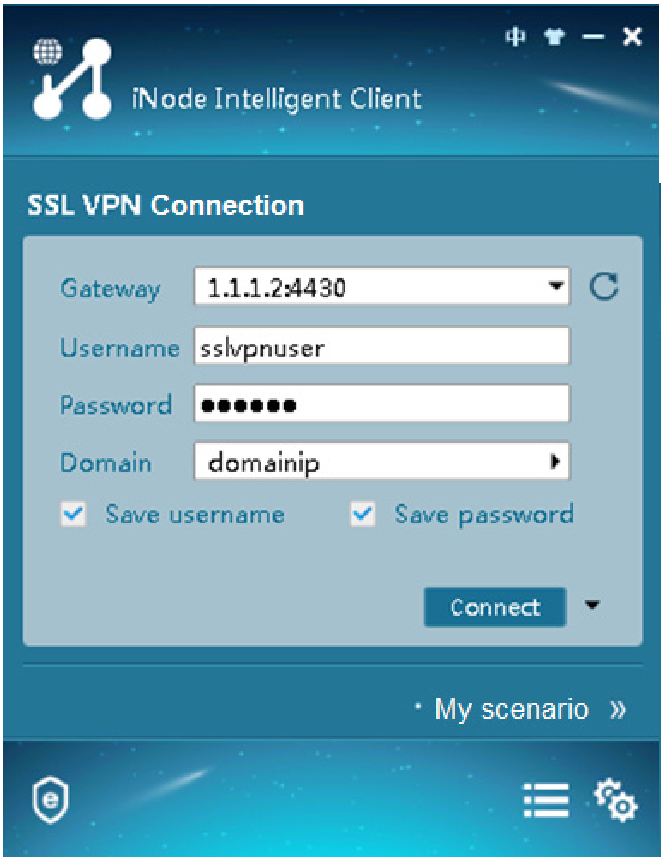
Username: sslvpnuser

Password: ••••••

# On the SSL VPN home page that opens, click **Start** to download the IP client application and install the application.

After the IP client application is installed, start the iNode client, as shown in Figure 22.

Figure 22 Starting the iNode client



iNode Intelligent Client

SSL VPN Connection

Gateway 1.1.1.2:4430

Username sslvpnuser

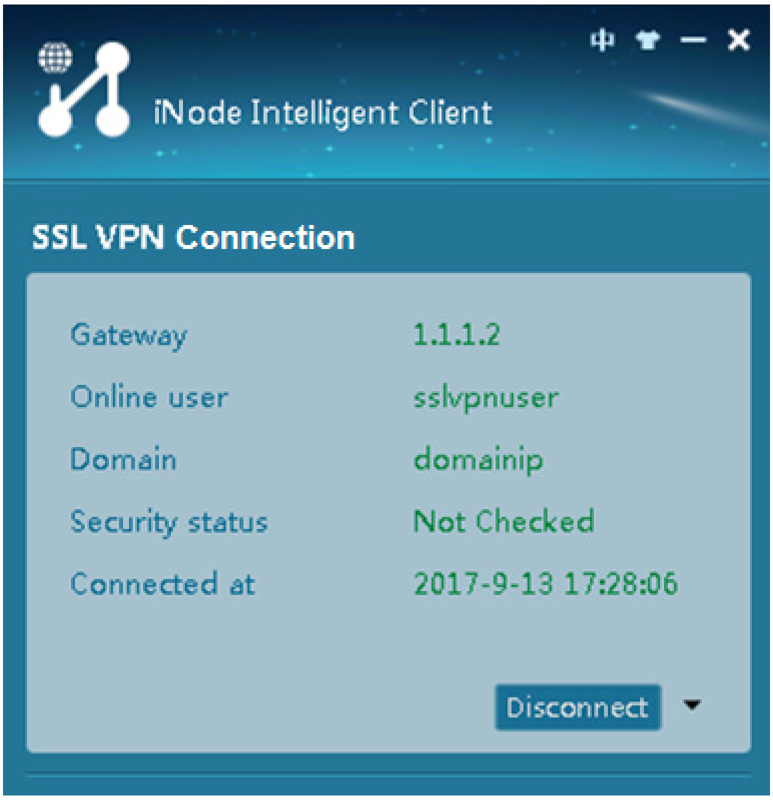
Password ●●●●●●

Domain domainip

☑ Save username ☑ Save password

# Click **Connect** to log in to the SSL VPN client, as shown in Figure 23.

Figure 23 Logging in to the SSL VPN client



Gateway 1.1.1.2
Online user sslvpnuser
Domain domainip
Security status Not Checked
Connected at 2017-9-13 17:28:06

# Verify that the user can ping the server.

```
C:\>ping 20.2.2.2
Pinging 20.2.2.2 with 32 bytes of data:
Reply from 20.2.2.2: bytes=32 time=31ms TTL=254
Reply from 20.2.2.2: bytes=32 time=18ms TTL=254
Reply from 20.2.2.2: bytes=32 time=15ms TTL=254
Reply from 20.2.2.2: bytes=32 time=16ms TTL=254
Ping statistics for 20.2.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 15ms, Maximum = 31ms, Average = 20ms
```

# Display SSL VPN session information on the device.

```
[Device] display sslvpn session user sslvpnuser
User              : sslvpnuser
Context           : ctxip
Policy group      : resourcegrp
Idle timeout      : 30 min
Created at        : 16:38:48 UTC Wed 07/26/2017
Lastest           : 16:47:41 UTC Wed 07/26/2017
User IPv4 address : 172.16.1.16
Allocated IP      : 10.1.1.1
Session ID        : 14
Web browser/OS    : Windows
```

Example: Configuring RADIUS authentication and authorization

Network configuration

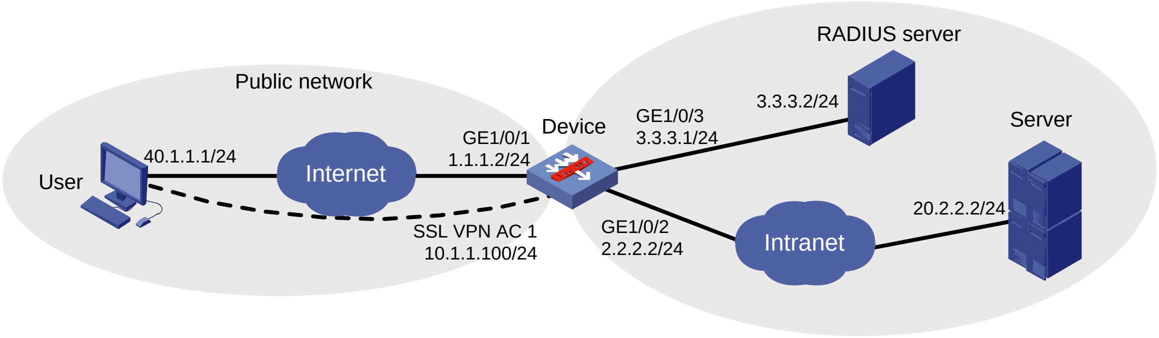
As shown in Figure 24, the device acts as an SSL VPN gateway that connects the public network and private network VPN 1.

The device uses a CA-signed SSL server certificate. If no SSL server policy is applied to the device, the device uses a self-signed SSL server certificate.

Configure SSL VPN IP access on the device to allow the user to access the internal server in the private network.

Configure the device to perform remote authentication and authorization (through the remote RADIUS server) for the user.

Figure 24 Network diagram



Prerequisites

Before configuring IP access, perform the following tasks:

- Make sure the server has a route to 10.1.1.0/24.
- Configure the RADIUS server to provide authentication and authorization for the user.

Procedure

1. Obtain CA certificate file **ca.cer** and local certificate file **server.pfx** for the device. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Create SSL VPN AC interface AC 1 and configure the IP address as 10.1.1.100/24 for the interface.

   ```
   [Device] interface sslvpn-ac 1
   [Device-SSLVPN-AC1] ip address 10.1.1.100 24
   [Device-SSLVPN-AC1] quit
   ```
4. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3 to the server, and 1.1.1.3 to the user.

```
[Device] ip route-static 20.2.2.2 24 2.2.2.3
[Device] ip route-static 40.1.1.1 24 1.1.1.3
```

5. Add interfaces to security zones.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] import interface sslvpn-ac 1
[Device-security-zone-Untrust] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] import interface gigabitethernet 1/0/3
[Device-security-zone-Trust] quit
```

6. Configure rules in a security policy to permit the traffic between security zones for the user to access the SSL VPN gateway and the server:

# Configure a rule named **sslvpnlocalout1** to permit the packets from the device to the user.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name sslvpnlocalout1
[Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
[Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 10.1.1.100
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-1-sslvpnlocalout1] action pass
[Device-security-policy-ip-1-sslvpnlocalout1] quit
```

# Configure a rule named **sslvpnlocalin1** to permit the packets from the user to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin1
[Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
[Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host subnet 10.1.1.0 24
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 10.1.1.100
[Device-security-policy-ip-2-sslvpnlocalin1] action pass
[Device-security-policy-ip-2-sslvpnlocalin1] quit
```

# Configure a rule named **sslvpnlocalout2** to permit the packets from the device to the server.

```
[Device-security-policy-ip] rule name sslvpnlocalout2
[Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 3.3.3.1
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 3.3.3.2
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

# Configure a rule named **sslvpnlocalin2** to permit the packets from the server to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
```

```
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 3.3.3.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 3.3.3.1
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
```

# Configure a rule named **untrust-trust** to allow the user to access the server through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-5-untrust-trust] source-zone untrust
[Device-security-policy-ip-5-untrust-trust] destination-zone trust
[Device-security-policy-ip-5-untrust-trust] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-5-untrust-trust] destination-ip-host 20.2.2.2
[Device-security-policy-ip-5-untrust-trust] action pass
[Device-security-policy-ip-5-untrust-trust] quit
```

# Configure a rule named **trust-untrust** to permit the packets from the server to the user through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-6-trust-untrust] source-zone trust
[Device-security-policy-ip-6-trust-untrust] destination-zone untrust
[Device-security-policy-ip-6-trust-untrust] source-ip-host 20.2.2.2
[Device-security-policy-ip-6-trust-untrust] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-6-trust-untrust] action pass
[Device-security-policy-ip-6-trust-untrust] quit
[Device-security-policy-ip] quit
```

7. Configure a PKI domain named **sslvpn** and certificate-related parameters.

```
<Device> system-view
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check  enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

8. Create an SSL server policy named **ssl** and specify PKI domain **sslvpn** for the policy.

```
[Device] ssl server-policy ssl
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] quit
```

9. Configure the SSL VPN gateway for user access. Configure the IP address for SSL VPN gateway **gw** as 1.1.1.2 and port number as 2000, and then apply server policy **ssl** to the gateway.

```
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 2000
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

10. Create an IP access address pool named **ippool** and specify the address range as 10.1.1.1 to 10.1.1.10.

```
[Device] sslvpn ip address-pool ippool 10.1.1.1 10.1.1.10
```

11. Configure RADIUS settings:

# Create a RADIUS scheme named **rscheme**. Specify the primary authentication server and primary accounting server as **3.3.3.2**. Set the keys for communication with the servers to **123456**.

```
[Device] radius scheme rscheme
[Device-radius-rscheme] primary authentication 3.3.3.2
[Device-radius-rscheme] primary accounting 3.3.3.2
[Device-radius-rscheme] accounting-on enable
[Device-radius-rscheme] key authentication simple 123456
[Device-radius-rscheme] key accounting simple 123456
```

# Exclude the domain name from the username sent to the RADIUS server.

```
[Device-radius-rscheme] user-name-format without-domain
[Device-radius-rscheme] quit
```

12. Create a user group named **group1** and authorize the user group to use SSL VPN policy group **pgroup**.

```
[Device] user-group group1
[Device-ugroup-group1] authorization-attribute sslvpn-policy-group pgroup
[Device-ugroup-group1] quit
```

13. Configure ISP domain **domain1**:

# Create an ISP domain named **domain1** and authorize the domain to use user group **group1**.

```
[Device] domain domain1
[Device-isp-domain1] authorization-attribute user-group group1
```

# Configure the ISP domain to use RADIUS scheme **rscheme** for AAA of users.

```
[Device-isp-domain1] authentication sslvpn radius-scheme rscheme
[Device-isp-domain1] authorization sslvpn radius-scheme rscheme
[Device-isp-domain1] accounting sslvpn radius-scheme rscheme
[Device-isp-domain1] quit
```

14. Create an SSL VPN context named **ctx**, specify gateway **gw** for the context, and then associate the context with VPN instance **VPN1**.

```
[Device] sslvpn context ctx
[Device-sslvpn-context-ctx] gateway gw
[Device-sslvpn-context-ctx] vpn-instance VPN1
[Device-sslvpn-context-ctx] aaa domain domain1
[Device-sslvpn-context-ctx] ip-route-list rtlist
[Device-sslvpn-context-ctx-route-list-rtlist] include 20.2.2.0 255.255.255.0
[Device-sslvpn-context-ctx-route-list-rtlist] quit
[Device-sslvpn-context-ctx] uri-acl uriacl
[Device-sslvpn-context-ctx-uri-acl-uriacl] rule 1 permit uri icmp://20.2.2.0
[Device-sslvpn-context-ctx-uri-acl-uriacl] quit
[Device-sslvpn-context-ctx] ip-tunnel interface sslvpn-ac 1
[Device-sslvpn-context-ctx] ip-tunnel address-pool ippool mask 255.255.255.0
[Device-sslvpn-context-ctx] policy-group pgroup
[Device-sslvpn-context-ctx-policy-group-pgroup] ip-tunnel access-route
ip-route-list rtlist
[Device-sslvpn-context-ctx-policy-group-pgroup] filter ip-tunnel uri-acl uriacl
[Device-sslvpn-context-ctx-policy-group-pgroup] quit
[Device-sslvpn-context-ctx] service enable
[Device-sslvpn-context-ctx] quit
```

Verifying the configuration

# Verify that SSL VPN gateway **gw** is up on the device.

```
[Device] display sslvpn gateway
Gateway name: gw
  Operation state: Up
  IP: 1.1.1.2  Port: 2000
  SSL server policy configured: ssl
  SSL server policy in use: ssl
  Front VPN instance: Not configured
```

# Verify that SSL VPN context **ctx** is up on the device.

```
[Device] display sslvpn context
Context name: ctx
  Operation state: Up
  AAA domain: domain1
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: Not configured
  Associated SSL VPN gateway: gw
  SSL client policy configured: ssl
  SSL client policy in use: ssl
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled
```

# On the user PC, launch the IP access client software, and then enter the address **1.1.1.2**, port number **2000**, username **sslvpn**, and password **123456** to log in to the SSL VPN gateway. (Details not shown.)

# Display SSL VPN session information on the device.

```
[Device] display sslvpn session context ctx
SSL VPN context: ctx
Users: 1
Username        Connections  Idle time   Created       User IP
sslvpn          6            0/00:02:05  0/00:03:14    40.1.1.1
```

# On the user PC, display IPv4 routing table to verify that the user has a route to the server.

NOTE:

The address 40.1.1.1/24 is the address of the local NIC, and 10.1.1.1/24 is the address that the SSL VPN gateway allocates to the user.

```
>route -4 print
IPv4 Route Table
===========================================================================
Active Routes:
```

```
Network Destination        Netmask          Gateway       Interface  Metric
         10.1.1.0    255.255.255.0         On-link      10.1.1.1        276
         10.1.1.1  255.255.255.255         On-link      10.1.1.1        276
       10.1.1.255  255.255.255.255         On-link      10.1.1.1        276
         20.2.2.0    255.255.255.0         On-link      10.1.1.1        276
       20.2.2.255  255.255.255.255         On-link      10.1.1.1        276
         40.1.1.0    255.255.255.0         On-link      40.1.1.1        276
         40.1.1.1  255.255.255.255         On-link      40.1.1.1        276
       40.1.1.255  255.255.255.255         On-link      40.1.1.1        276
===========================================================================
```

# Verify that the user can ping the server.

```
C:\>ping 20.2.2.2
Pinging 20.2.2.2 with 32 bytes of data:
Reply from 20.2.2.2: bytes=32 time=197ms TTL=254
Reply from 20.2.2.2: bytes=32 time=1ms TTL=254
Reply from 20.2.2.2: bytes=32 time=1ms TTL=254
Reply from 20.2.2.2: bytes=32 time=186ms TTL=254

Ping statistics for 20.2.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 197ms, Average = 96ms
```

Example: Configuring LDAP authentication and authorization

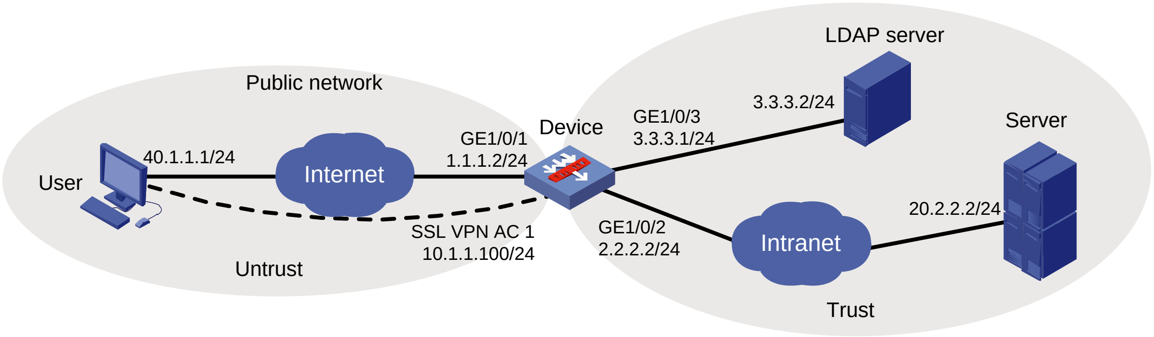
Network configuration

As shown in Figure 25, the device acts as an SSL VPN gateway. The SSL VPN gateway IP address is 1.1.1.2 and the service port number is 8080.

The device uses a CA-signed SSL server certificate. If no SSL server policy is applied to the device, the device uses a self-signed SSL server certificate.

Use an LDAP server to perform authentication and authorization for SSL VPN users. The LDAP server runs Microsoft Windows Server 2008 R2 Active Directory and uses domain **ldap.com**. The server assigns an SSL VPN policy group named **pgroup** to an SSL VPN user after the user passes authentication. The policy group specifies the Web resources that the user can access.

Figure 25 Network diagram



Configuring the LDAP server

1. Add an organizational unit named **sslvpn_usergroup**:

a. On the LDAP server, select **Start** > **Bandizip** > **Administrative Tools**.

b. Double-click **Active Directory Users and Computers**.

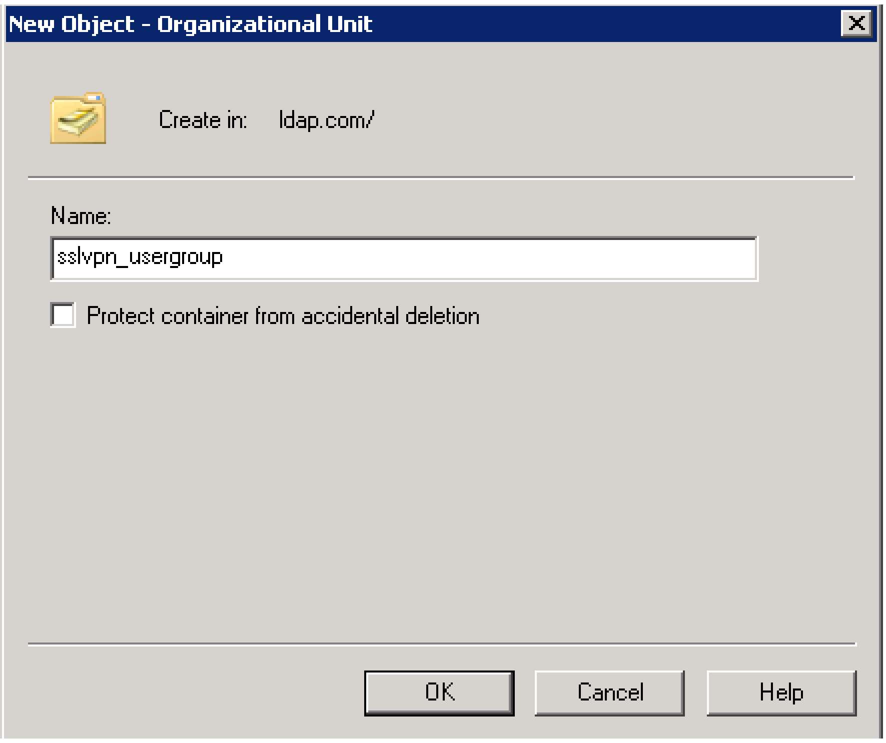
The **Active Directory Users and Computers** window opens.

c. From the navigation tree, right-click **ldap.com**.

d. Select **New** > **Organizational Unit** from the menu to display the dialog box for adding an organizational unit.

e. Enter organizational unit name **sslvpn_usergroup** and click **OK**.

Figure 26 Adding organizational unit sslvpn_usergroup



New Object - Organizational Unit

Create in: ldap.com/

Name: sslvpn_usergroup

☐ Protect container from accidental deletion

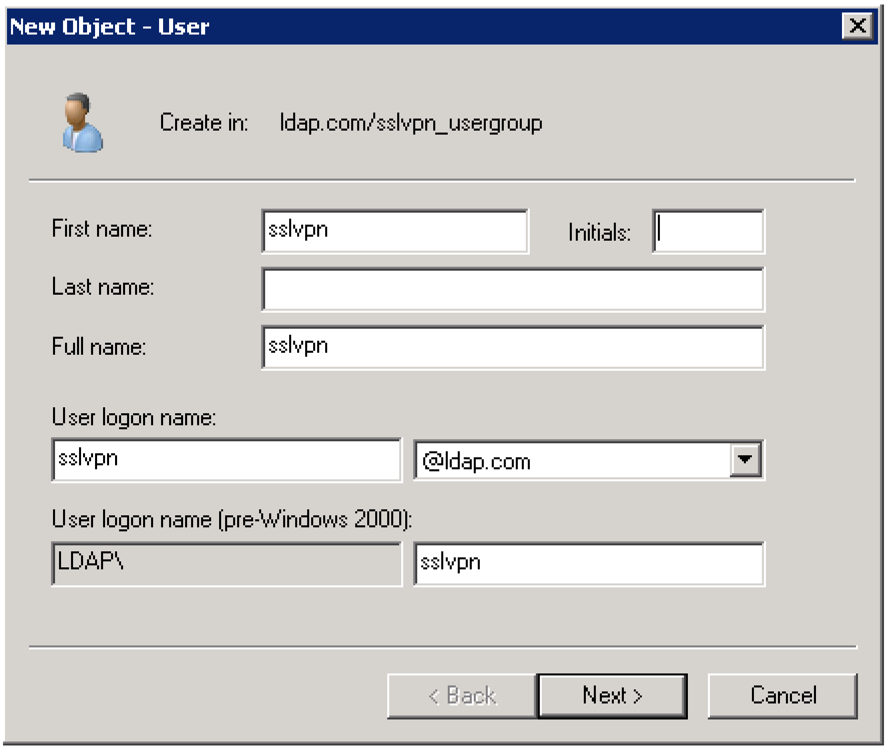
2. Add a user named **sslvpn** under organizational unit **sslvpn_usergroup** and set the password to **ldap!123456**:

a. From the navigation tree, right-click **sslvpn_usergroup**.

b. Select **New** > **User** from the menu to display the dialog box for adding a user.

c. Enter logon name **sslvpn** and click **Next**.

Figure 27 Adding user sslvpn



Create in: ldap.com/sslvpn_usergroup

First name: sslvpn

Initials:

Last name:

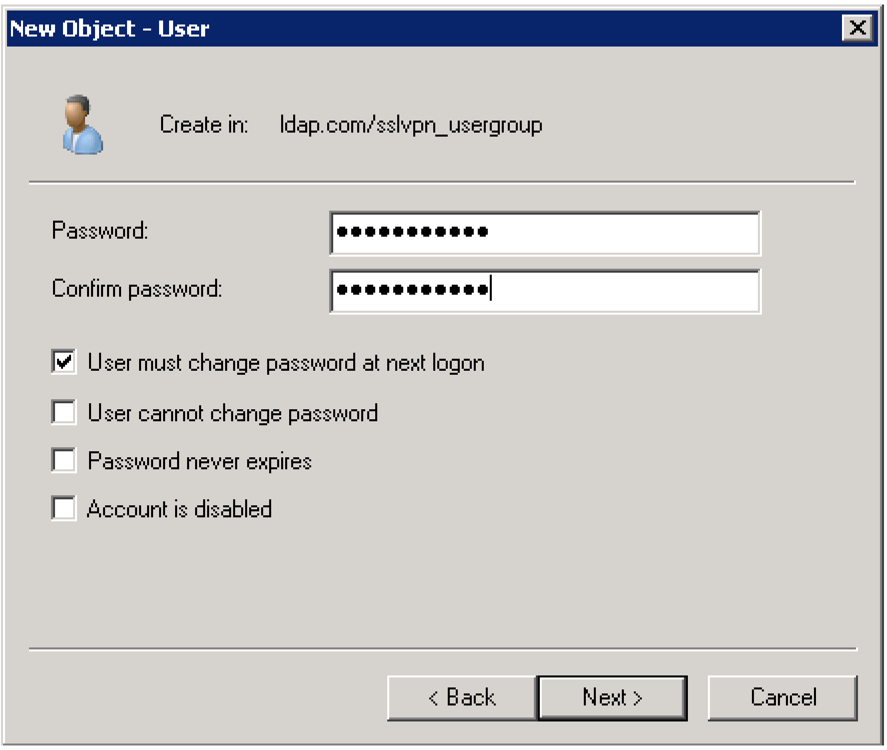
Full name: sslvpn

User logon name: sslvpn @ldap.com

User logon name (pre-Windows 2000): LDAP\ sslvpn

a. In the dialog box, enter password **ldap!123456**, select options as needed, and click **Next**.

Figure 28 Setting the user's password



Create in: ldap.com/sslvpn_usergroup

Password: ●●●●●●●●●●●

Confirm password: ●●●●●●●●●●●

☑ User must change password at next logon

□ User cannot change password

□ Password never expires

□ Account is disabled

a. Click **Next**.

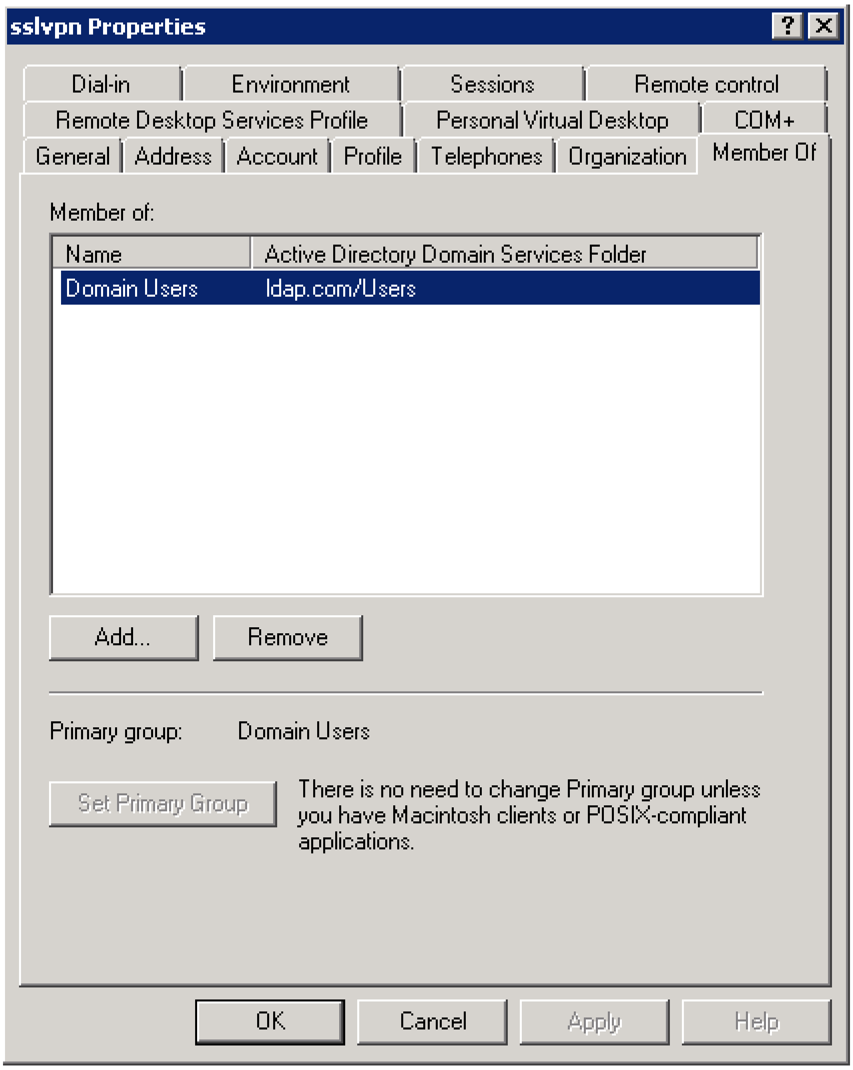
3. Add user **sslvpn** to group **Users**:

a. From the navigation tree, click **sslvpn_usergroup**.

b. In the right pane, right-click user **sslvpn** and select **Properties**.

c. In the dialog box, click the **Member Of** tab and click **Add**.

Figure 29 Modifying user properties



Member of:

Name	Active Directory Domain Services Folder
Domain Users	ldap.com/Users

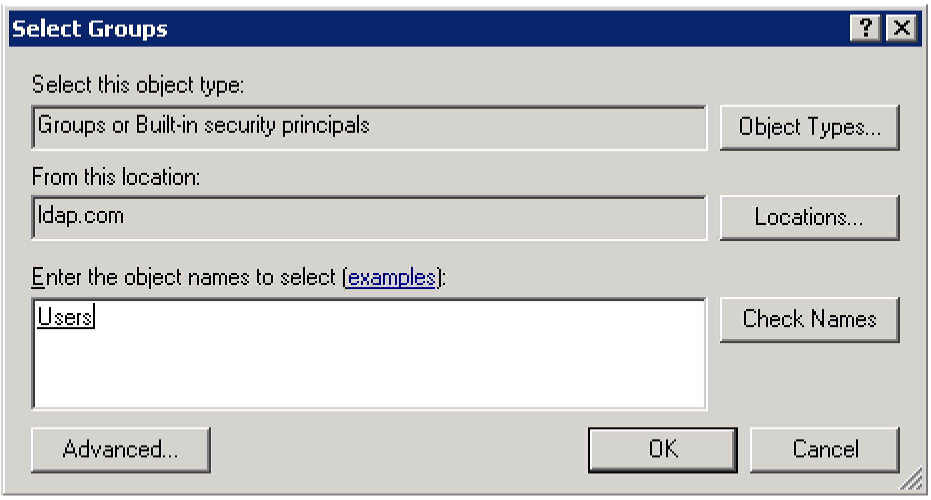
Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

a. In the **Select Groups** dialog box, enter **Users** in the **Enter the object names to select** field, and click **OK**.

User **sslvpn** is added to group **Users**.

Figure 30 Adding user sslvpn to group Users



Select Groups

Select this object type:

Groups or Built-in security principals

From this location:

ldap.com

Enter the object names to select (examples):

Users

Configuring the device

1. Obtain CA certificate file **ca.cer** and local certificate file **server.pfx** for the device. (Details not shown.)
2. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)
3. Create SSL VPN AC interface AC 1 and configure the IP address as 10.1.1.100/24 for the interface.

   ```
   [Device] interface sslvpn-ac 1
    [Device-SSLVPN-AC1] ip address 10.1.1.100 24
   [Device-SSLVPN-AC1] quit
   ```
4. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3 to the server, and 1.1.1.3 to the user.

   ```
   [Device] ip route-static 20.2.2.2 24 2.2.2.3
   [Device] ip route-static 40.1.1.1 24 1.1.1.3
   ```
5. Add interfaces to security zones.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] import interface sslvpn-ac 1
   [Device-security-zone-Untrust] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/3
   [Device-security-zone-Trust] quit
   ```
6. Configure rules in a security policy to permit the traffic between security zones for the user to access the SSL VPN gateway and the server:

   # Configure a rule named **sslvpnlocalout1** to permit the packets from the device to the user.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name sslvpnlocalout1
[Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
[Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
[Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 10.1.1.100
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
[Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-1-sslvpnlocalout1] action pass
[Device-security-policy-ip-1-sslvpnlocalout1] quit
```

# Configure a rule named **sslvpnlocalin1** to permit the packets from the user to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin1
[Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
[Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
[Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host subnet 10.1.1.0 24
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
[Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 10.1.1.100
[Device-security-policy-ip-2-sslvpnlocalin1] action pass
[Device-security-policy-ip-2-sslvpnlocalin1] quit
```

# Configure a rule named **sslvpnlocalout2** to permit the packets from the device to the server.

```
[Device-security-policy-ip] rule name sslvpnlocalout2
[Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 3.3.3.1
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 3.3.3.2
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

# Configure a rule named **sslvpnlocalin2** to permit the packets from the server to the device.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 3.3.3.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 3.3.3.1
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
```

# Configure a rule named **untrust-trust** to allow the user to access the server through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-5-untrust-trust] source-zone untrust
[Device-security-policy-ip-5-untrust-trust] destination-zone trust
[Device-security-policy-ip-5-untrust-trust] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-5-untrust-trust] destination-ip-host 20.2.2.2
[Device-security-policy-ip-5-untrust-trust] action pass
```

```
[Device-security-policy-ip-5-untrust-trust] quit
```

# Configure a rule named **trust-untrust** to permit the packets from the server to the user through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-6-trust-untrust] source-zone trust
[Device-security-policy-ip-6-trust-untrust] destination-zone untrust
[Device-security-policy-ip-6-trust-untrust] source-ip-host 20.2.2.2
[Device-security-policy-ip-6-trust-untrust] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-6-trust-untrust] action pass
[Device-security-policy-ip-6-trust-untrust] quit
[Device-security-policy-ip] quit
```

7. Configure a PKI domain named **sslvpn** and certificate-related parameters.

```
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check  enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

8. Create an SSL server policy named **ssl** and specify PKI domain **sslvpn** for the policy.

```
[Device] ssl server-policy ssl
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] quit
```

9. Configure the SSL VPN gateway for user access. Configure the IP address for SSL VPN gateway **gw** as 1.1.1.2 and port number as 2000, and then apply server policy **ssl** to the gateway.

```
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 2000
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

10. Create an IP access address pool named **ippool** and specify the address range as 10.1.1.1 to 10.1.1.10.

```
[Device] sslvpn ip address-pool ippool 10.1.1.1 10.1.1.10
```

11. Configure LDAP settings for SSL VPN user authentication.

```
[Device] ldap server ldap1
[Device-ldap-server-ldap1] ip 3.3.3.2
[Device-ldap-server-ldap1] login-dn cn=admin,cn=users,dc=ldap,dc=com
[Device-ldap-server-ldap1] login-password simple admin!123456
[Device-ldap-server-ldap1] search-base-dn dc=ldap,dc=com
[Device-ldap-server-ldap1] quit
[Device] ldap attribute-map test
[Device-ldap-attr-map-test] map ldap-attribute memberof prefix cn= delimiter , 
aaa-attribute user-group
[Device-ldap-attr-map-test] quit
[Device] ldap scheme shml
[Device-ldap-shml] authentication-server ldap1
[Device-ldap-shml] authorization-server ldap1
[Device-ldap-shml] attribute-map test
[Device-ldap-shml] quit
```

12. Create an ISP domain named **bbb** and configure the authentication, authorization, and accounting methods for SSL VPN users.

```
[Device] domain bbb
[Device-isp-bbb] authentication sslvpn ldap-scheme shml
[Device-isp-bbb] authorization sslvpn ldap-scheme shml
[Device-isp-bbb] accounting sslvpn none
[Device-isp-bbb] quit
```

13. Create an SSL VPN context named **ctx**, specify gateway **gw** for the context, and then associate the context with VPN instance **VPN1**.

```
[Device] sslvpn context ctx
[Device-sslvpn-context-ctx] gateway gw
[Device-sslvpn-context-ctx] vpn-instance VPN1
[Device-sslvpn-context-ctx] aaa domain bbb
[Device-sslvpn-context-ctx] ip-route-list rtlist
[Device-sslvpn-context-ctx-route-list-rtlist] include 20.2.2.0 255.255.255.0
[Device-sslvpn-context-ctx-route-list-rtlist] quit
[Device-sslvpn-context-ctx] uri-acl uriacl
[Device-sslvpn-context-ctx-uri-acl-uriacl] rule 1 permit uri icmp://20.2.2.0
[Device-sslvpn-context-ctx-uri-acl-uriacl] quit
[Device-sslvpn-context-ctx] ip-tunnel interface sslvpn-ac 1
[Device-sslvpn-context-ctx] ip-tunnel address-pool ippool mask 255.255.255.0
[Device-sslvpn-context-ctx] policy-group pgroup
[Device-sslvpn-context-ctx-policy-group-pgroup] ip-tunnel access-route
ip-route-list rtlist
[Device-sslvpn-context-ctx-policy-group-pgroup] filter ip-tunnel uri-acl uriacl
[Device-sslvpn-context-ctx-policy-group-pgroup] quit
[Device-sslvpn-context-ctx] service enable
[Device-sslvpn-context-ctx] quit
```

14. Create a user group named **users** and authorize the user group to use SSL VPN policy group **pgroup**.

```
[Device] user-group users
[Device-ugroup-users] authorization-attribute sslvpn-policy-group pgroup
[Device-ugroup-users] quit
```

Verifying the configuration

# Verify that SSL VPN gateway **gw** is up on the device.

```
[Device] display sslvpn gateway
Gateway name: gw
  Operation state: Up
  IP: 1.1.1.2  Port: 2000
  SSL server policy configured: ssl
  SSL server policy in use: ssl
  Front VPN instance: Not configured
```

# Verify that SSL VPN context **ctx** is up on the device.

```
[Device] display sslvpn context
Context name: ctx
  Operation state: Up
  AAA domain: domain1
  Certificate authentication: Disabled
  Password authentication: Enabled
  Authentication use: All
  SMS auth type: Not configured
  Urlmasking: Disabled
  Code verification: Disabled
  Default policy group: Not configured
  Associated SSL VPN gateway: gw
  SSL client policy configured: ssl
  SSL client policy in use: ssl
  Maximum users allowed: 1048575
  VPN instance: Not configured
  Idle timeout: 30 min
  Authentication server-type: aaa
  Password changing: Enabled
```

# On the user PC, launch the IP access client software, and then enter the address **1.1.1.2**, port number **2000**, username **sslvpn**, and password **123456** to log in to the SSL VPN gateway. (Details not shown.)

# Display SSL VPN session information on the device.

```
[Device] display sslvpn session context ctx
SSL VPN context: ctx
Users: 1
Username        Connections  Idle time   Created       User IP
sslvpn          6            0/00:02:05  0/00:03:14    40.1.1.1
```

# On the user PC, display IPv4 routing table to verify that the user has a route to the server.

NOTE:

The address 40.1.1.1/24 is the address of the local NIC, and 10.1.1.1/24 is the address that the SSL VPN gateway allocates to the user.

```
>route -4 print
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
         10.1.1.0    255.255.255.0         On-link      10.1.1.1        276
         10.1.1.1  255.255.255.255         On-link      10.1.1.1        276
       10.1.1.255  255.255.255.255         On-link      10.1.1.1        276
         20.2.2.0    255.255.255.0         On-link      10.1.1.1        276
       20.2.2.255  255.255.255.255         On-link      10.1.1.1        276
         40.1.1.0    255.255.255.0         On-link      40.1.1.1        276
         40.1.1.1  255.255.255.255         On-link      40.1.1.1        276
       40.1.1.255  255.255.255.255         On-link      40.1.1.1        276
===========================================================================
```

# Verify that the user can ping the server.

```
C:\>ping 20.2.2.2
Pinging 20.2.2.2 with 32 bytes of data:
Reply from 20.2.2.2: bytes=32 time=197ms TTL=254
Reply from 20.2.2.2: bytes=32 time=1ms TTL=254
Reply from 20.2.2.2: bytes=32 time=1ms TTL=254
Reply from 20.2.2.2: bytes=32 time=186ms TTL=254

Ping statistics for 20.2.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 197ms, Average = 96ms
```

Example: Configuring IP access with USB key certificate authentication

Network configuration

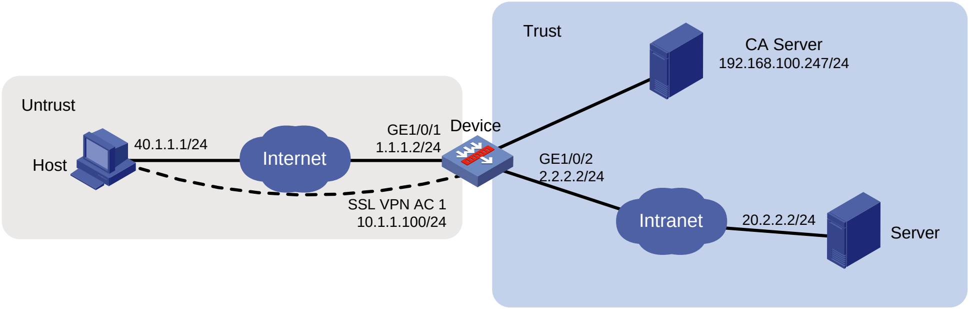
As shown in Figure 31, the device acts as an SSL VPN gateway that connects the public network and the private network.

Configure SSL VPN IP access on the device to allow a user on the public network to access the internal server on the private network securely.

Configure the device to perform certificate-based identity authentication for the user and authorize the user to access the internal server after the user passes the authentication.

The user uses a USB key to log in to the SSL VPN gateway.

Figure 31 Network diagram



Restrictions and guidelines

The device can use a self-signed SSL server certificate or a CA-signed SSL server certificate.

- **Self-signed SSL server certificate**—Factory default server certificate on the device. If the device uses a self-signed SSL server certificate, no SSL server policy is needed.
- **CA-signed SSL server certificate**—User requested server certificate. If the device uses a CA-signed SSL server certificate, you must specify an SSL server policy.

Because the self-signed SSL server certificate is not secure, use it only for a functional test. In practice, please use a CA-signed SSL server certificate.

Prerequisites

Before configuring SSL VPN, make sure of the following:

- The device has obtained CA certificate **ca.cer** and server certificate **server.pfx**. The USB key has installed certificates. The certificates in the USB key and those on the device are issued by the same CA.
- The specified attribute (CN attribute by default) in the client certificate of the USB key is the same as the username of the SSL VPN user.
- Install the driver for the USB key to ensure availability of the USB key.

Configuring the SSL VPN gateway device

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Create an SSL VPN AC interface and assign an IP address to it to forward the IP access traffic.

   ```
   [Device] interface sslvpn-ac 1
   [Device-SSLVPN-AC1] ip address 10.1.1.100 24
   [Device-SSLVPN-AC1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] import interface sslvpn-ac 1
   [Device-security-zone-Untrust] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```

3. Configure settings for routing.

   This example configures static routes, and the next hop in the route to the server is 2.2.2.3, and the next hop in the route to the host is 1.1.1.3.

   ```
   [Device] ip route-static 20.2.2.2 24 2.2.2.3
   [Device] ip route-static 40.1.1.1 24 1.1.1.3
   ```

4. Configure security policy rules to permit traffic between security zones for the user to access the SSL VPN gateway and the server.

   # Configure a rule named **sslvpnlocalout1** to allow the SSL VPN gateway to send packets to the user.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name sslvpnlocalout1
   [Device-security-policy-ip-1-sslvpnlocalout1] source-zone local
   [Device-security-policy-ip-1-sslvpnlocalout1] destination-zone untrust
   [Device-security-policy-ip-1-sslvpnlocalout1] source-ip-host 1.1.1.2
   [Device-security-policy-ip-1-sslvpnlocalout1] destination-ip-host 40.1.1.1
   [Device-security-policy-ip-1-sslvpnlocalout1] action pass
   [Device-security-policy-ip-1-sslvpnlocalout1] quit
   ```

   # Configure a rule named **sslvpnlocalin1** to allow the user to send packets to the SSL VPN gateway.

   ```
   [Device-security-policy-ip] rule name sslvpnlocalin1
   [Device-security-policy-ip-2-sslvpnlocalin1] source-zone untrust
   [Device-security-policy-ip-2-sslvpnlocalin1] destination-zone local
   [Device-security-policy-ip-2-sslvpnlocalin1] source-ip-host 40.1.1.1
   [Device-security-policy-ip-2-sslvpnlocalin1] destination-ip-host 1.1.1.2
   [Device-security-policy-ip-2-sslvpnlocalin1] action pass
   [Device-security-policy-ip-2-sslvpnlocalin1] quit
   ```

# Configure a rule named **sslvpnlocalout2** to allow the SSL VPN gateway to send packets to the server.

```
[Device-security-policy-ip] rule name sslvpnlocalout2
[Device-security-policy-ip-3-sslvpnlocalout2] source-zone local
[Device-security-policy-ip-3-sslvpnlocalout2] destination-zone trust
[Device-security-policy-ip-3-sslvpnlocalout2] source-ip-host 2.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] destination-ip-host 20.2.2.2
[Device-security-policy-ip-3-sslvpnlocalout2] action pass
[Device-security-policy-ip-3-sslvpnlocalout2] quit
```

# Configure a rule named **sslvpnlocalin2** to allow the server to send packets to the SSL VPN gateway.

```
[Device-security-policy-ip] rule name sslvpnlocalin2
[Device-security-policy-ip-4-sslvpnlocalin2] source-zone trust
[Device-security-policy-ip-4-sslvpnlocalin2] destination-zone local
[Device-security-policy-ip-4-sslvpnlocalin2] source-ip-host 20.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] destination-ip-host 2.2.2.2
[Device-security-policy-ip-4-sslvpnlocalin2] action pass
[Device-security-policy-ip-4-sslvpnlocalin2] quit
```

# Configure a rule named **untrust-trust** to allow the user to access the server through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-5-untrust-trust] source-zone untrust
[Device-security-policy-ip-5-untrust-trust] destination-zone trust
[Device-security-policy-ip-5-untrust-trust] source-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-5-untrust-trust] destination-ip-host 20.2.2.2
[Device-security-policy-ip-5-untrust-trust] action pass
[Device-security-policy-ip-5-untrust-trust] quit
```

# Configure a rule named **trust-untrust** to allow the server to send packets to the user through the SSL VPN AC interface.

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-6-trust-untrust] source-zone trust
[Device-security-policy-ip-6-trust-untrust] destination-zone untrust
[Device-security-policy-ip-6-trust-untrust] source-ip-host 20.2.2.2
[Device-security-policy-ip-6-trust-untrust] destination-ip-subnet 10.1.1.0 24
[Device-security-policy-ip-6-trust-untrust] action pass
[Device-security-policy-ip-6-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure certificates for the device, which are used by the SSL VPN client to authenticate the SSL VPN gateway.

# Configure a PKI domain to import the certificates.

```
[Device] pki domain sslvpn
[Device-pki-domain-sslvpn] public-key rsa general name sslvpn
[Device-pki-domain-sslvpn] undo crl check  enable
[Device-pki-domain-sslvpn] quit
[Device] pki import domain sslvpn der ca filename ca.cer
[Device] pki import domain sslvpn p12 local filename server.pfx
```

# Configure an SSL VPN server policy, specify the PKI domain, and enable client authentication in the policy.

```
[Device] ssl server-policy ssl
```

```
[Device-ssl-server-policy-ssl] pki-domain sslvpn
[Device-ssl-server-policy-ssl] client-verify enable
[Device-ssl-server-policy-ssl] quit
```

6. Configure SSL VPN:

   # Configure the SSL VPN gateway.

```
[Device] sslvpn gateway gw
[Device-sslvpn-gateway-gw] ip address 1.1.1.2 port 4430
[Device-sslvpn-gateway-gw] ssl server-policy ssl
[Device-sslvpn-gateway-gw] service enable
[Device-sslvpn-gateway-gw] quit
```

   # Create an SSL VPN client pool, which is used to assign IP addresses to IP access users.

```
[Device] sslvpn ip address-pool sslvpnpool 10.1.1.1 10.1.1.10
```

   # Create an ACL to filter IP access traffic.

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip source 10.1.1.0 0.0.0.255 destination
20.2.2.0 0.0.0.255
[Device-acl-ipv4-adv-3000] quit
```

   # Configure an SSL VPN context, enable certificate authentication for users, and provide SSL VPN IP access services to users.

```
[Device] sslvpn context ctxip
[Device-sslvpn-context-ctxip] gateway gw
[Device-sslvpn-context-ctxip] certificate-authentication enable
[Device-sslvpn-context-ctxip] ip-tunnel interface sslvpn-ac 1
[Device-sslvpn-context-ctxip] ip-route-list rtlist
[Device-sslvpn-context-ctxip-route-list-rtlist] include 20.2.2.0 24
[Device-sslvpn-context-ctxip-route-list-rtlist] quit
[Device-sslvpn-context-ctxip] ip-tunnel address-pool sslvpnpool mask 24
[Device-sslvpn-context-ctxip] policy-group resourcegrp
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] ip-tunnel access-route
ip-route-list rtlist
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] filter ip-tunnel acl 3000
[Device-sslvpn-context-ctxip-policy-group-resourcegrp] quit
[Device-sslvpn-context-ctxip] service enable
[Device-sslvpn-context-ctxip] quit
```

7. Configure an SSL VPN user, which is used to access the SSL VPN gateway.

   # Create a local SSL VPN user named **sslvpnuser**, specify the password as **123456TESTplat&!**, user role as **network-operator**, and the SSL VPN policy group authorized as **resourcegrp**.

```
[Device] local-user sslvpnuser class network
[Device-luser-network-sslvpnuser] password simple 123456TESTplat&!
[Device-luser-network-sslvpnuser] service-type sslvpn
[Device-luser-network-sslvpnuser] authorization-attribute sslvpn-policy-group
resourcegrp
[Device-luser-network-sslvpnuser] authorization-attribute user-role
network-operator
[Device-luser-network-sslvpnuser] quit
```

Configuring the server

Make sure the server has a route to subnet 10.1.1.0/24.

Verifying the configuration

1. Display SSL VPN information on the device:

   # Display SSL VPN gateway information. The output shows that the SSL VPN gateway is in UP state.

   ```
   [Device] display sslvpn gateway
   Gateway name: gw
     Operation state: Up
     IP: 1.1.1.2  Port: 4430
     Front VPN instance: Not configured
   ```

   # Display SSL VPN context information. The output shows that the SSL VPN context is in UP state.

   ```
   [Device] display sslvpn context
   Context name: ctxip
     Operation state: Up
     AAA domain: Not specified
     Certificate authentication: Enabled
     Password authentication: Enabled
     Authentication use: All
     SMS auth type: Not configured
     Urlmasking: Disabled
     Code verification: Disabled
     Default policy group: Not configured
     Associated SSL VPN gateway: gw
     Maximum users allowed: 1048575
     VPN instance: Not configured
     Idle timeout: 30 min
     Authentication server-type: aaa
     Password changing: Enabled
   ```

   # After the user logs in to the SSL VPN gateway, display SSL VPN session information on the device. The output shows the session information of SSL VPN user **sslvpnuser**.

   ```
   [Device] display sslvpn session user sslvpnuser
   User              : sslvpnuser
   Context           : ctxip
   Policy group      : resourcegrp
   Idle timeout      : 30 min
   Created at        : 16:38:48 UTC Wed 07/26/2017
   Lastest           : 16:47:41 UTC Wed 07/26/2017
   User IPv4 address : 172.16.1.16
   Allocated IP      : 10.1.1.1
   Session ID        : 14
   Web browser/OS    : Windows
   ```

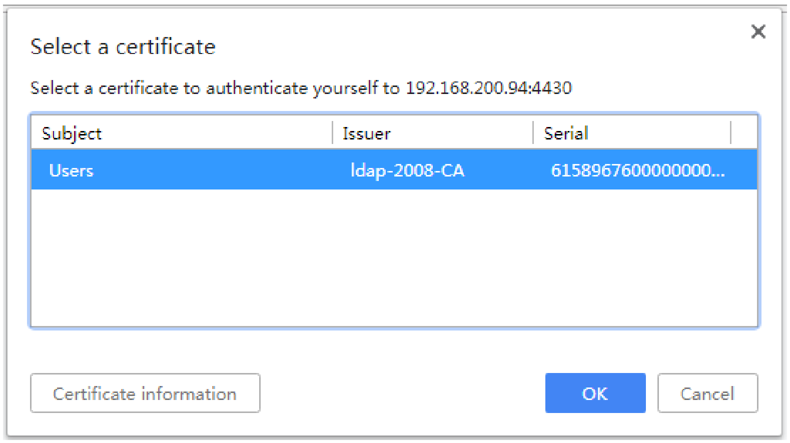
2. Install a USB key on the host.

   Obtain the USB key from the administrator, and install the USB key on the host. For information about how to make a USB key, see the appendix in the following section.

3. Log in to the SSL VPN gateway from the host:

   # On the host, type the gateway address **https://1.1.1.2:4430/** in the address bar of a browser, and then press **Enter**. The following page opens:

Figure 32 Selecting a certificate



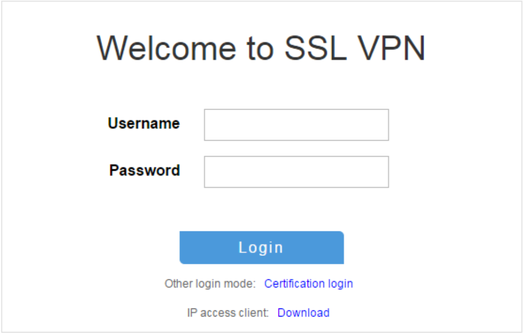
Select a certificate

Select a certificate to authenticate yourself to 192.168.200.94:4430

Subject	Issuer	Serial
Users	ldap-2008-CA	6158967600000000...

# Select a certificate, and then click **OK**. The SSL VPN login page opens.

Figure 33 Login page



Welcome to SSL VPN

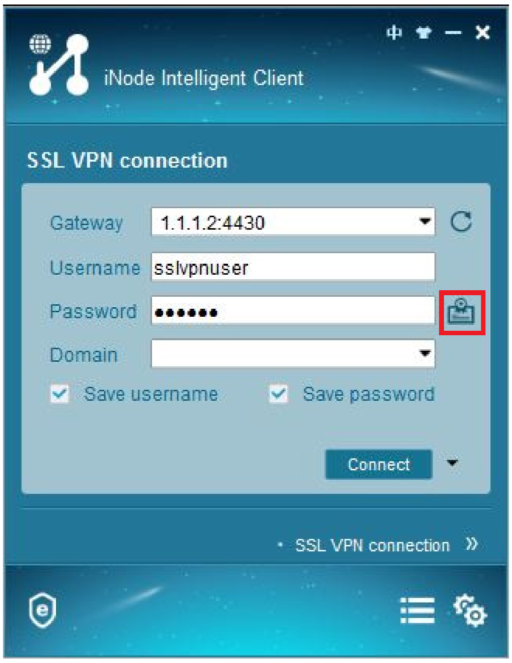
Username
Password
Login
Other login mode: Certification login
IP access client: Download

# Enter username **sslvpnuser** and password **123456TESTplat&!**, and then click **Login**.

# On the SSL VPN home page that opens, click **Start** to download the IP client application and install the application.

# Launch the installed IP client and configure it as follows:

Figure 34 iNode client



Gateway: 1.1.1.2:4430

Username: sslvpnuser

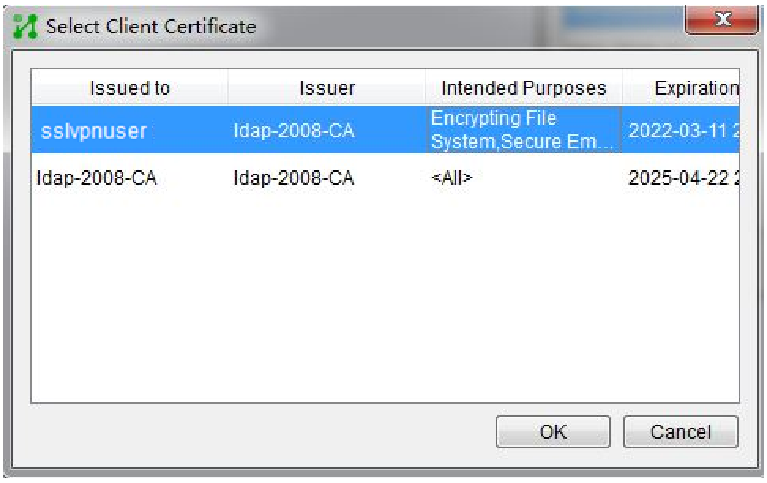
Password: ••••••

Domain:

☑ Save username ☑ Save password

# Click the icon next the **Password** box. In the dialog box that opens, select the client certificate in the USB key, and then click **OK**.

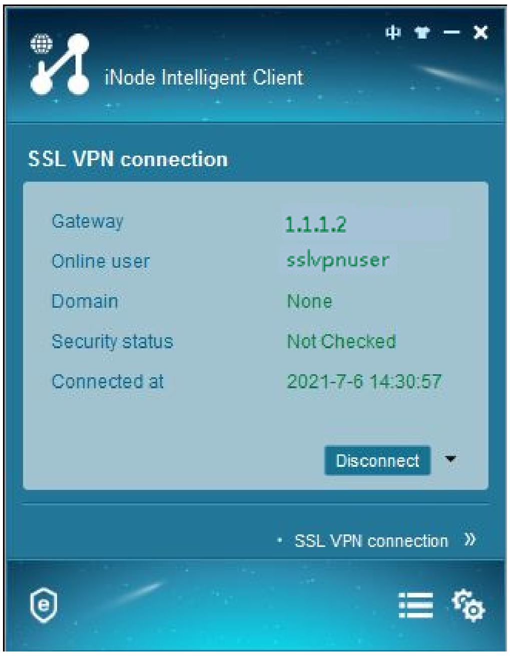
Figure 35 Selecting the client certificate



Issued to	Issuer	Intended Purposes	Expiration
sslvpnuser	ldap-2008-CA	Encrypting File System,Secure Em...	2022-03-11 2
ldap-2008-CA	ldap-2008-CA	<All>	2025-04-22 2

# Click **Connect** on the iNode client. You log in to the SSL VPN gateway successfully.

Figure 36 Logging into the SSL VPN gateway successfully



SSL VPN connection

Gateway 1.1.1.2

Online user sslvpnuser

Domain None

Security status Not Checked

Connected at 2021-7-6 14:30:57

# After the SSL VPN user logs in, the user can ping the server IP address 20.2.2.2 from the host.

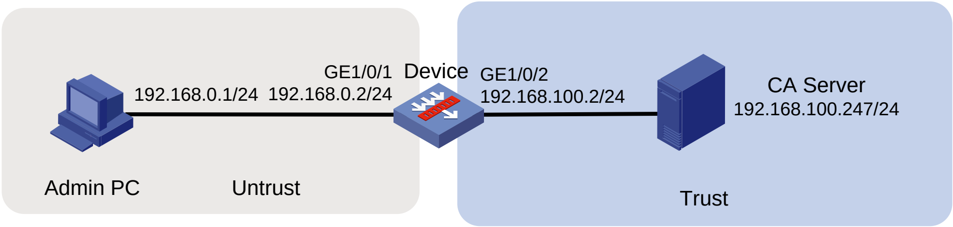
```
C:\>ping 20.2.2.2
Pinging 20.2.2.2 with 32 bytes of data:
Reply from 20.2.2.2: bytes=32 time=31ms TTL=254
Reply from 20.2.2.2: bytes=32 time=18ms TTL=254
Reply from 20.2.2.2: bytes=32 time=15ms TTL=254
Reply from 20.2.2.2: bytes=32 time=16ms TTL=254

Ping statistics for 20.2.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 15ms, Maximum = 31ms, Average = 20ms
```

Appendix—Making a USB key

Make a USB key in the following procedure:

1. Configure an IP address and gateway on the administrator's PC to ensure the PC can reach the CA server. This example uses Windows 2008 server as the CA server.

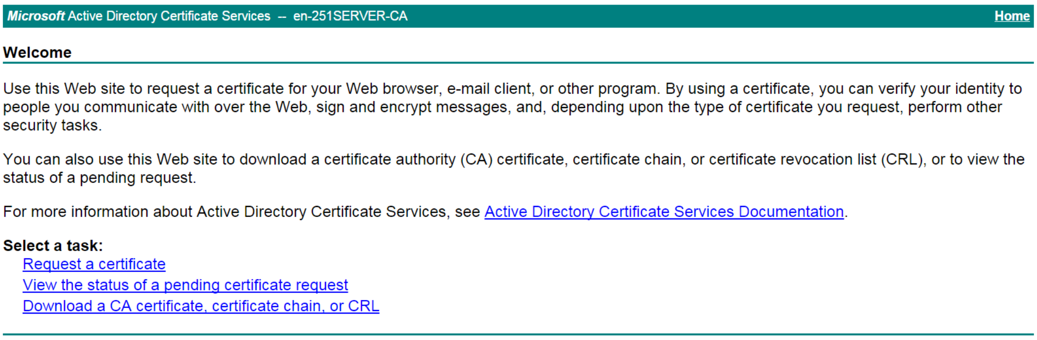
Figure 37 Network diagram



2. Request the USB key client certificate:

# Enter **http://192.168.100.247/certsrv** in the address bar of a browser to open the certificate service page.

Figure 38 Certificate services



Microsoft Active Directory Certificate Services -- en-251SERVER-CA

Welcome

Use this Web site to request a certificate for your Web browser, e-mail client, or other program. By using a certificate, you can verify your identity to people you communicate with over the Web, sign and encrypt messages, and, depending upon the type of certificate you request, perform other security tasks.

You can also use this Web site to download a certificate authority (CA) certificate, certificate chain, or certificate revocation list (CRL), or to view the status of a pending request.

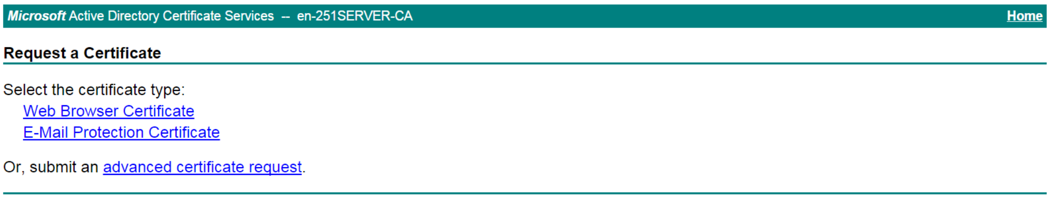
For more information about Active Directory Certificate Services, see Active Directory Certificate Services Documentation.

Select a task:

Request a certificate
View the status of a pending certificate request
Download a CA certificate, certificate chain, or CRL

# Click **Request a certificate**. The certificate request page opens.

Figure 39 Requesting a certificate



Microsoft Active Directory Certificate Services -- en-251SERVER-CA

Request a Certificate

Select the certificate type:
Web Browser Certificate
E-Mail Protection Certificate

Or, submit an advanced certificate request.

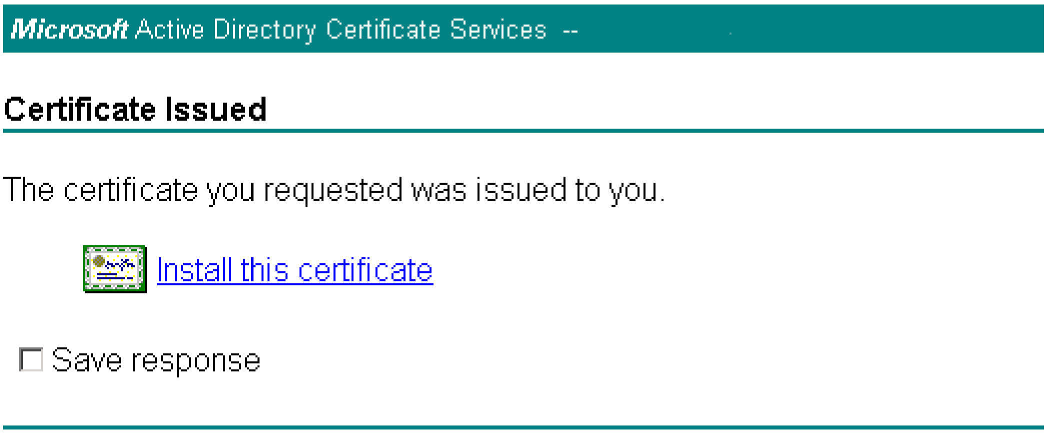
# Click **advanced certificate request**. On the page that opens, select **Create and submit a request to this CA** to request a client certificate.

# Configure the client certificate request parameters, and then click **Submit** at the bottom of the page.

# In the dialog box that opens, enter the USB key password, and then log in.

# Click **Install this certificate** to install the client certificate to the USB key.

Figure 40 Installing the client certificate to the USB key



Microsoft Active Directory Certificate Services --

Certificate Issued

The certificate you requested was issued to you.

Install this certificate

□ Save response

# After a possible conflict warning about installing a certificate, click **Yes** to install the client certificate into the USB key.

The USB key is made successfully.

Contents

Configuring IPsec

About IPsec

IP Security (IPsec) is defined by the IETF to provide interoperable, high-quality, cryptography-based security for IP communications. It is a Layer 3 VPN technology that transmits data in a secure channel established between two endpoints (such as two security gateways). Such a secure channel is usually called an IPsec tunnel.

IPsec framework

IPsec is a security framework that has the following protocols and algorithms:

- Authentication Header (AH).
- Encapsulating Security Payload (ESP).
- Internet Key Exchange (IKE).
- Algorithms for authentication and encryption.

AH and ESP are security protocols that provide security services. IKE performs automatic key exchange. For more information about IKE, see "Configuring IKE."

IPsec security services

IPsec provides the following security services for data packets in the IP layer:

- **Confidentiality**—The sender encrypts packets before transmitting them over the Internet, protecting the packets from being eavesdropped en route.
- **Data integrity**—The receiver verifies the packets received from the sender to make sure they are not tampered with during transmission.
- **Data origin authentication**—The receiver verifies the authenticity of the sender.
- **Anti-replay**—The receiver examines packets and drops outdated and duplicate packets.

Benefits of IPsec

IPsec delivers the following benefits:

- Reduced key negotiation overhead and simplified maintenance by supporting the IKE protocol. IKE provides automatic key negotiation and automatic IPsec security association (SA) setup and maintenance.
- Good compatibility. You can apply IPsec to all IP-based application systems and services without modifying them.
- Encryption on a per-packet rather than per-flow basis. Per-packet encryption allows for flexibility and greatly enhances IP security.

Security protocols

IPsec comes with two security protocols, AH and ESP. They define how to encapsulate IP packets and the security services that they can provide.

- AH (protocol 51) defines the encapsulation of the AH header in an IP packet, as shown in Figure 3. AH can provide data origin authentication, data integrity, and anti-replay services to prevent data tampering, but it cannot prevent eavesdropping. Therefore, it is suitable for transmitting non-confidential data. Authentication algorithms supported by AH include HMAC-MD5 and HMAC-SHA1. AH does not support NAT traversal.

- ESP (protocol 50) defines the encapsulation of the ESP header and trailer in an IP packet, as shown in Figure 3. ESP can provide data encryption, data origin authentication, data integrity, and anti-replay services. Unlike AH, ESP can guarantee data confidentiality because it can encrypt the data before encapsulating the data to IP packets. ESP-supported encryption algorithms include DES, 3DES, and AES, and authentication algorithms include HMAC-MD5 and HMAC-SHA1.

Both AH and ESP provide authentication services, but the authentication service provided by AH is stronger. In practice, you can choose either or both security protocols. When both AH and ESP are used, an IP packet is encapsulated first by ESP and then by AH.

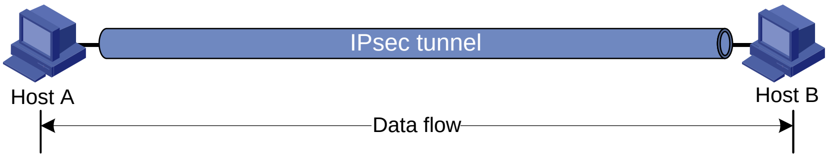
Encapsulation modes

IPsec supports the following encapsulation modes: transport mode and tunnel mode.

Transport mode

The security protocols protect the upper layer data of an IP packet. Only the transport layer data is used to calculate the security protocol headers. The calculated security protocol headers and the encrypted data (only for ESP encapsulation) are placed after the original IP header. You can use the transport mode when end-to-end security protection is required (the secured transmission start and end points are the actual start and end points of the data). The transport mode is typically used for protecting host-to-host communications, as shown in Figure 1.

Figure 1 IPsec protection in transport mode



Tunnel mode

The security protocols protect the entire IP packet. The entire IP packet is used to calculate the security protocol headers. The calculated security protocol headers and the encrypted data (only for ESP encapsulation) are encapsulated in a new IP packet. In this mode, the encapsulated packet has two IP headers. The inner IP header is the original IP header. The outer IP header is added by the network device that provides the IPsec service. You must use the tunnel mode when the secured transmission start and end points are not the actual start and end points of the data packets (for example, when two gateways provide IPsec but the data start and end points are two hosts behind the gateways). The tunnel mode is typically used for protecting gateway-to-gateway communications, as shown in Figure 2.

Figure 2 IPsec protection in tunnel mode

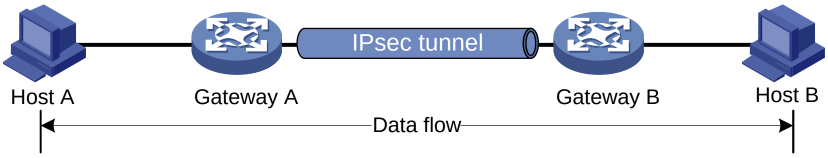
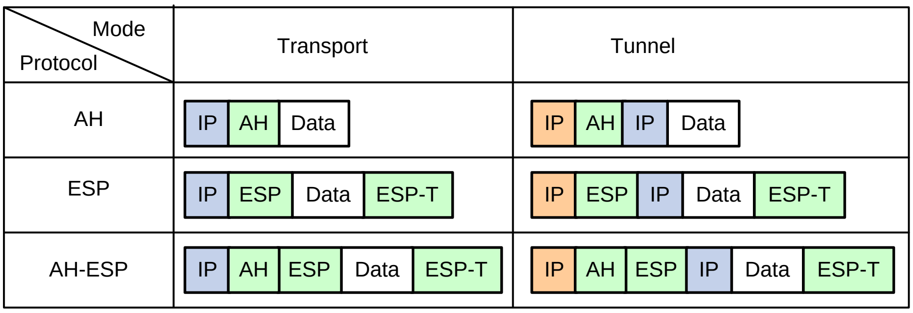


Figure 3 shows how the security protocols encapsulate an IP packet in different encapsulation modes.

Figure 3 Security protocol encapsulations in different modes



Mode / Protocol	Transport	Tunnel
AH	IP | AH | Data	IP | AH | IP | Data
ESP	IP | ESP | Data | ESP-T	IP | ESP | IP | Data | ESP-T
AH-ESP	IP | AH | ESP | Data | ESP-T	IP | AH | ESP | IP | Data | ESP-T

Security association

About this task

A security association (SA) is an agreement negotiated between two communicating parties called IPsec peers. An SA includes the following parameters for data protection:

- Security protocols (AH, ESP, or both).
- Encapsulation mode (transport mode or tunnel mode).
- Authentication algorithm (HMAC-MD5, SM3, or HMAC-SHA1).
- Encryption algorithm (DES, 3DES, SM, or AES).
- Shared keys and their lifetimes.

An SA is unidirectional. At least two SAs are needed to protect data flows in a bidirectional communication. If two peers want to use both AH and ESP to protect data flows between them, they construct an independent SA for each protocol in each direction.

An SA is uniquely identified by a triplet, which consists of the security parameter index (SPI), destination IP address, and security protocol identifier. An SPI is a 32-bit number. It is transmitted in the AH/ESP header.

SA setup

An SA can be set up manually or through IKE.

- **Manual mode**—Configure all parameters for the SA through commands. This configuration mode is complex and does not support some advanced features (such as periodic key update), but it can implement IPsec without IKE. This mode is mainly used in small and static networks or when the number of IPsec peers in the network is small.
- **IKE negotiation mode**—The peers negotiate and maintain the SA through IKE. This configuration mode is simple and has good expansibility. As a best practice, set up SAs through IKE negotiations in medium- and large-scale dynamic networks.

SA aging

A manually configured SA never ages out.

An IKE-created SA has a lifetime and will be deleted when its lifetime timer expires.

Before the SA lifetime timer expires, IKE negotiates a new SA, which takes over immediately after its creation. The interval from the creation of an SA to the negotiation of a new SA is the SA's soft lifetime.

The SA soft lifetime is calculated as follows: SA soft lifetime = SA lifetime – SA soft lifetime buffer. If the SA soft lifetime buffer is not configured, the system calculates a default SA soft lifetime based on the SA lifetime.

The lifetime of an IKE-created SA comes in two types:

- **Time-based lifetime**—Defines how long the SA can exist after it is created.
- **Traffic-based lifetime**—Defines the maximum traffic that the SA can process.

If both lifetime timers are configured for an SA, the SA is deleted when either of the lifetime timers expires.

Authentication and encryption

Authentication algorithms

IPsec uses hash algorithms to perform authentication. A hash algorithm produces a fixed-length digest for an arbitrary-length message. IPsec peers respectively calculate message digests for each packet. The receiver compares the local digest with that received from the sender. If the digests are identical, the receiver considers the packet intact and the sender's identity valid. IPsec supports the following types of authentication algorithms:

- Hash-based Message Authentication Code (HMAC) based authentication algorithms, including HMAC-MD5 and HMAC-SHA.

  HMAC-MD5 is faster but less secure than HMAC-SHA.
- SM3 authentication algorithms.

Encryption algorithms

IPsec uses symmetric encryption algorithms, which encrypt and decrypt data by using the same keys. The following encryption algorithms are available for IPsec on the device:

- **DES**—Encrypts a 64-bit plaintext block with a 56-bit key. DES is the least secure but the fastest algorithm.
- **3DES**—Encrypts plaintext data with three 56-bit DES keys. The key length totals up to 168 bits. It provides moderate security strength and is slower than DES.
- **AES**—Encrypts plaintext data with a 128-bit, 192-bit, or 256-bit key. AES provides the highest security strength and is slower than 3DES.
- **SM**—Encrypts plaintext data with a 128-bit key. SM provides the same level of security strength as AES.

Crypto engine

The IPsec feature is resource intensive for its complex encryption/decryption and authentication algorithms. To improve processing performance, you can use crypto engine to offload IPsec tasks.

The crypto engine processes all IPsec protected packets and hands the processed packets back to the device for forwarding.

For more information about crypto engines, see "Configuring crypto engines."

The hardware key pairs stored on a GM crypto card become invalid after the GM crypto card is removed from a device and installed on a new device.

IPsec-protected traffic

IPsec tunnels can protect the following types of traffic:

- Packets that match specific ACLs.
- Packets routed to a tunnel interface.
- Packets of IPv6 routing protocols.

Two peers use security policies (IPsec policies or IPsec profiles) to protect packets between them. A security policy defines the range of packets to be protected by IPsec and the security parameters used for the protection. For more information about IPsec policies and IPsec profiles, see "IPsec policy and IPsec profile."

The following information describes how IPsec protects packets:

- When an IPsec peer identifies the packets to be protected according to the security policy, it sets up an IPsec tunnel and sends the packet to the remote peer through the tunnel. The IPsec tunnel can be manually configured beforehand, or it can be set up through IKE negotiation triggered by the packet. The IPsec tunnels are actually the IPsec SAs. The inbound packets are protected by the inbound SA, and the outbound packets are protected by the outbound SA.
- When the remote IPsec peer receives the packet, it drops, de-encapsulates, or directly forwards the packet according to the configured security policy.

ACL-based IPsec

To implement ACL-based IPsec, configure an ACL to define the data flows to be protected, specify the ACL in an IPsec policy, and then apply the IPsec policy to an interface. You can apply an IPsec policy to physical interfaces such as Ethernet interfaces, or virtual interfaces such as tunnel interfaces and virtual template interfaces.

ACL-based IPsec works as follows:

- When packets sent by the interface match a permit rule of the ACL, the packets are protected by the outbound IPsec SA and encapsulated with IPsec.
- When the interface receives an IPsec packet destined for the local device, it searches for the inbound IPsec SA according to the SPI in the IPsec packet header for de-encapsulation. If the de-encapsulated packet matches a permit rule of the ACL, the device processes the packet. If the de-encapsulated packet does not match any permit rule of the ACL, the device drops the packet.

The device supports the following data flow protection modes:

- **Standard mode**—One IPsec tunnel protects one data flow. The data flow permitted by an ACL rule is protected by one IPsec tunnel that is established solely for it.
- **Aggregation mode**—One IPsec tunnel protects all data flows permitted by all the rules of an ACL. This mode is only used to communicate with old-version devices.
- **Per-host mode**—One IPsec tunnel protects one host-to-host data flow. One host-to-host data flow is identified by one ACL rule and protected by one IPsec tunnel established solely for it. This mode consumes more system resources when multiple data flows exist between two subnets to be protected.

Tunnel interface-based IPsec

Tunnel interface-based IPsec is also known as virtual tunnel interface (VTI)-based IPsec.

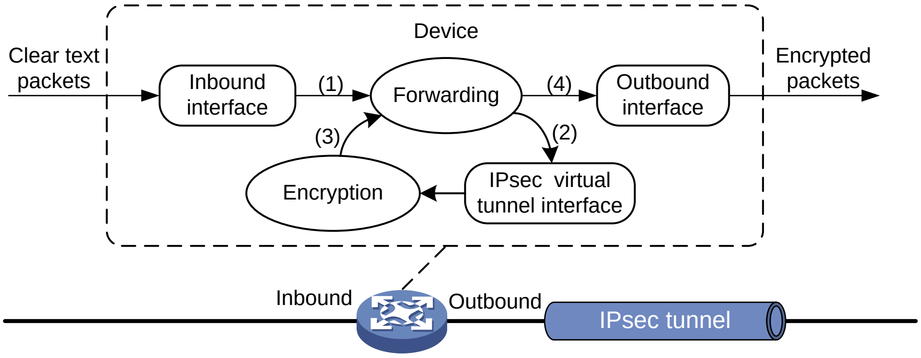
To implement tunnel interface-based IPsec, configure an IPsec profile and apply the IPsec profile to a tunnel interface. IPsec will protect all traffic routed to the tunnel interface, except the traffic that you specify not to protect. Tunnel interface-based IPsec supports only the tunnel encapsulation mode.

Compared with ACL-based IPsec, tunnel interface-based IPsec has the following advantages:

- Supports multicast traffic protection.
- Supports dynamic routing protocol advertisement between the IPsec tunnel peers.
- Simplifies configuration. Tunnel interface-based IPsec does not require using ACL rules to define the traffic to be protected. The routing table directs the traffic to the tunnel interface for protection.

For tunnel interface-based IPsec, packet encapsulation and decapsulation are performed on the tunnel interfaces.

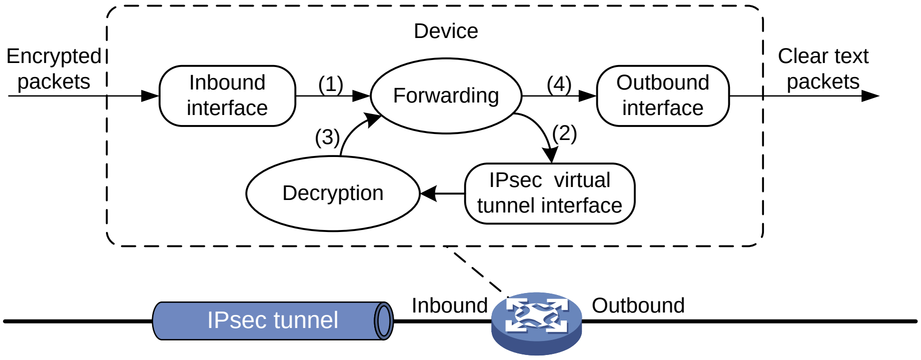
Figure 4 Tunnel interface encapsulation



As shown in Figure 4, a tunnel interface encapsulates an IP packet as follows:

1. Upon receiving a clear text packet, the input interface sends the packet to the forwarding module for routing.
2. If the packet requires IPsec protection, the forwarding module sends the packet to the tunnel interface.
3. The tunnel interface encapsulates the packet into a new IP packet. The source and destination IP addresses in the new IP header are the source and destination IP addresses of the tunnel interface. Then, the tunnel interface sends the packet back to the forwarding module.
4. The forwarding module looks up the routing table again and sends the packet out of the physical interface of the tunnel interface.

Figure 5 Tunnel interface de-encapsulation



As shown in Figure 5, a tunnel interface de-encapsulates an IP packet as follows:

1. Upon receiving an encrypted packet, the inbound interface sends the packet to the forwarding module for routing.
2. Because the packet is destined for the tunnel interface' source address and the payload protocol is AH or ESP, the forwarding module sends the packet to the tunnel interface.
3. The tunnel interface de-encapsulates the packet (removes the outer IP header) and sends the de-encapsulated packet back to the forwarding module.
4. The forwarding module looks up the routing table again and sends the packet out of the output interface.

IPv6 routing protocol-based IPsec

You can implement IPv6 routing protocol-based IPsec by binding an IPsec profile to an IPv6 routing protocol. All packets of the protocol are encapsulated with IPsec. Supported IPv6 routing protocols include OSPFv3, IPv6 BGP, and RIPng.

All packets of the applications that are not bound to IPsec and the IPsec packets that failed to be de-encapsulated are dropped.

In one-to-many communication scenarios, you must configure the IPsec SAs for an IPv6 routing protocol in manual mode because of the following reasons:

- The automatic key exchange mechanism protects communications between two points. In one-to-many communication scenarios, automatic key exchange cannot be implemented.
- One-to-many communication scenarios require that all the devices use the same SA parameters (SPI and key) to receive and send packets. IKE negotiated SAs cannot meet this requirement.

IPsec policy and IPsec profile

IPsec policies and IPsec profiles define the parameters used to establish IPsec tunnels between two peers and the range of packets to be protected.

IPsec policy

An IPsec policy is a set of IPsec policy entries that have the same name but different sequence numbers.

An IPsec policy contains the following settings:

- An ACL that defines the range of data flows to be protected.
- An IPsec transform set that defines the security parameters used for IPsec protection.
- IPsec SA establishment mode.

  Supported IPsec SA establishment modes are manual configuration and IKE negotiation.
- Local and remote IP addresses that define the start and end points of the IPsec tunnel.

In the same IPsec policy, an IPsec policy entry with a smaller sequence number has a higher priority. When sending a packet, the interface applied with an IPsec policy looks through the IPsec policy's entries in ascending order of sequence numbers. If the packet matches the ACL of an IPsec policy entry, the interface encapsulates the packet according to the IPsec policy entry. If no match is found, the interface sends the packet out without IPsec protection.

When the interface receives an IPsec packet destined for the local device, it searches for the inbound IPsec SA according to the SPI in the IPsec packet header for de-encapsulation. If the de-encapsulated packet matches a permit rule of the ACL, the device processes the packet. If the de-encapsulated packet does not match a permit rule of the ACL, the device drops the packet.

To protect traffic by using IPsec, you must apply an IPsec policy to an interface. The interface can be a physical interface, such as an Ethernet interface. It can also be a virtual interface, such as a tunnel and virtual template interface, to protect applications such as GRE and L2TP.

IPsec profile

An IPsec profile has similar settings as an IPsec policy. It is uniquely identified by a name and does not support ACL configuration.

IPsec profiles can be classified into the following types:

- **Manual IPsec profile**—A manual IPsec profile is used to protect IPv6 routing protocols. It specifies the IPsec transform set used for protecting data flows, and the SPIs and keys used by the SAs.
- **IKE-based IPsec profile**—An IKE-based IPsec profile is applied to tunnel interfaces to protect tunneled traffic. It specifies the IPsec transform sets used for protecting data flows, and the IKE profile used for IKE negotiation.

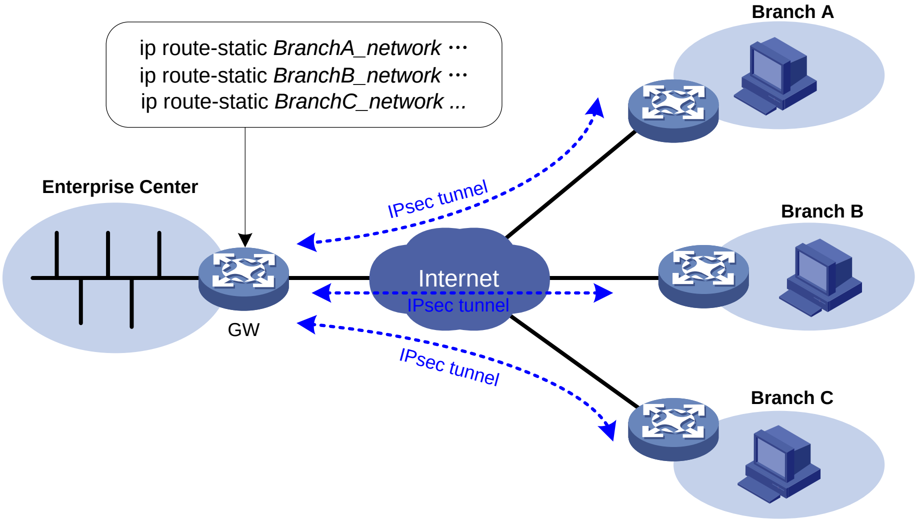
IPsec RRI

IPsec Reverse Route Injection (RRI) enables an IPsec tunnel gateway to automatically add and delete static routes destined for the protected private networks. It automatically adds the static routes when the IPsec SAs are established and deletes the static routes when the IPsec SAs are deleted. This greatly reduces the static route configuration work load on the gateway and increases the scalability of the IPsec VPN.

IPsec RRI is applicable to gateways that must provide many IPsec tunnels (for example, a headquarters gateway).

As shown in Figure 6, the traffic between the enterprise center and the branches are protected by IPsec. The gateway at the enterprise center is configured with static routes to route traffic to the IPsec-protected interfaces. It is difficult to add or modify static routes on the gateway at the enterprise center if the IPsec VPN has a large number of branches or if the network structure changes.

Figure 6 IPsec VPN


ip route-static *BranchA_network* ···
ip route-static *BranchB_network* ···
ip route-static *BranchC_network* ...
Branch A
Enterprise Center
IPsec tunnel
Branch B
Internet
IPsec tunnel
GW
IPsec tunnel
Branch C

After you can enable IPsec RRI on the gateway, the gateway automatically adds a static route to the routing table each time an IPsec tunnel is established. The destination IP address is the protected private network. The next hop IP address can be the remote IP address of the IPsec tunnel (default) or a user-defined next hop IP address. Traffic destined for the peer end is routed to the IPsec tunnel interface and thereby protected by IPsec.

You can advertise the static routes created by IPsec RRI in the internal network, and the internal network device can use them to forward traffic in the IPsec VPN.

You can set preferences for the static routes created by IPsec RRI to implement flexible route management. For example, you can set the same preference for multiple routes to the same destination to implement load sharing, or you can set different preferences to implement route backup.

You can also set tags for the static routes created by IPsec RRI to implement flexible route control through routing policies.

IPsec smart link selection

To improve network stability and availability, a branch's IPsec gateway typically deploys multiple links to connect to the corporate headquarters. The qualities of these links (in terms of packet loss ratio and delay) are not static but keep changing with time. It is important that the branch gateway can dynamically select a link with desired transmission quality to establish the IPsec tunnel to the headquarters. IPsec smart link selection can meet this requirement.

IPsec smart link selection enables the branch gateway to monitor the real-time packet loss ratio and delay of the active link over which the IPsec tunnel is established. If the packet loss ratio or delay of the link exceeds the specified threshold, IPsec smart link selection reselects a link for the IPsec tunnel. You can also manually activate a link to establish the IPsec tunnel over that link.

Protocols and standards

- RFC 2401, *Security Architecture for the Internet Protocol*
- RFC 2402, *IP Authentication Header*
- RFC 2406, *IP Encapsulating Security Payload*
- RFC 4552, *Authentication/Confidentiality for OSPFv3*

Restrictions: Hardware compatibility with IPsec

The SM1 algorithm is supported only on devices installed with a GM network data encryption module.

Restrictions and guidelines: IPsec configuration

Typically, IKE uses UDP port 500 for communication, and AH and ESP use the protocol numbers 51 and 50, respectively. Make sure traffic of these protocols is not denied on the interfaces with IKE or IPsec configured.

Implementing ACL-based IPsec

In transport mode, ACLs for IPsec take effect only on traffic that is generated by the device and traffic that is destined for the device. They do not take effect on traffic forwarded through the device. For example, an ACL-based IPsec tunnel can protect log messages the device sends to a log server, but it does not protect data flows and voice flows that are forwarded by the device.

ACL-based IPsec tasks at a glance

To configure ACL-based IPsec, perform the following tasks:

1. Configuring an ACL
2. Configuring an IPsec transform set
3. Configuring an IPsec policy

   Choose one of the following tasks:

   - Configuring a manual IPsec policy
   - Configuring an IKE-based IPsec policy
4. Applying an IPsec policy to an interface
5. (Optional.) Configuring accessibility features for ACL-based IPsec
   - Enabling ACL checking for de-encapsulated packets
   - Configuring IPsec anti-replay
   - Configuring IPsec anti-replay redundancy
   - Binding a source interface to an IPsec policy

   - Enabling QoS pre-classify
   - Configuring the DF bit of IPsec packets
   - Configuring IPsec RRI
   - Configuring IPsec smart link selection
   - Configuring IPsec netmask filtering
   - Enabling IPsec flow overlap check
   - Configuring the global IPsec SA lifetime and idle timeout
   - Configuring IPsec fragmentation
   - Setting the maximum number of IPsec tunnels
6. (Optional.) Configuring logging and SNMP notification for IPsec.
   - Enabling logging for IPsec packets
   - Enabling logging for IPsec negotiation
   - Configuring SNMP notifications for IPsec

Configuring an ACL

IPsec uses ACLs to identify the traffic to be protected.

Keywords in ACL rules

An ACL is a collection of ACL rules. Each ACL rule is a deny or permit statement. A permit statement identifies a data flow protected by IPsec, and a deny statement identifies a data flow that is not protected by IPsec. IPsec compares a packet against the ACL rules and processes the packet according to the first rule it matches.

- Each ACL rule matches both the outbound traffic and the returned inbound traffic. Suppose there is a rule **rule 0 permit ip source 1.1.1.0 0.0.0.255 destination 2.2.2.0 0.0.0.255**. This rule matches both traffic from 1.1.1.0 to 2.2.2.0 and traffic from 2.2.2.0 to 1.1.1.0.
- In the outbound direction, if a permit statement is matched, IPsec considers that the packet requires protection and continues to process it. If a deny statement is matched or no match is found, IPsec considers that the packet does not require protection and delivers it to the next module.
- In the inbound direction:
  - Non-IPsec packets that match a permit statement are dropped.
  - IPsec packets destined for the device itself are de-encapsulated. By default, the de-encapsulated packets are compared against the ACL rules. Only those that match a permit statement are processed. Other packets are dropped. If ACL checking for de-encapsulated IPsec packets is disabled, the de-encapsulated packets are not compared against the ACL rules and are directly processed by other modules.

When defining ACL rules for IPsec, follow these guidelines:

- Permit only data flows that need to be protected and use the **any** keyword with caution. With the **any** keyword specified in a permit statement, all outbound traffic matching the permit statement will be protected by IPsec. All inbound IPsec packets matching the permit statement will be received and processed, but all inbound non-IPsec packets will be dropped. This will cause all the inbound traffic that does not need IPsec protection to be dropped.
- Avoid statement conflicts in the scope of IPsec policy entries. When creating a deny statement, be careful with its match scope and match order relative to permit statements. The policy entries in an IPsec policy have different match priorities. ACL rule conflicts between them are prone to cause mistreatment of packets. For example, when configuring a permit statement for an IPsec policy entry to protect an outbound traffic flow, you must avoid the situation that the traffic flow matches a deny statement in a higher priority IPsec policy entry. Otherwise, the

packets will be sent out as normal packets. If they match a permit statement at the receiving end, they will be dropped by IPsec.

The following example shows how an improper statement causes unexpected packet dropping. Only the ACL-related configuration is presented.

Assume Device A is connected to subnet 1.1.2.0/24 and Device B is connected to subnet 3.3.3.0/24, and the IPsec policy configuration on Device A and Device B is as follows:

- IPsec configuration on Device A:

```
acl advanced 3000
 rule 0 permit ip source 1.1.1.0 0.0.0.255 destination 2.2.2.0 0.0.0.255
 rule 1 deny ip
acl advanced 3001
 rule 0 permit ip source 1.1.2.0 0.0.0.255 destination 3.3.3.0 0.0.0.255
 rule 1 deny ip
#
ipsec policy testa 1 isakmp <---IPsec policy entry with a higher priority
 security acl 3000
 ike-profile aa
 transform-set 1
#
ipsec policy testa 2 isakmp <---IPsec policy entry with a lower priority
 security acl 3001
 ike-profile bb
 transform-set 1
```

- IPsec configuration on Device B:

```
acl advanced 3001
 rule 0 permit ip source 3.3.3.0 0.0.0.255 destination 1.1.2.0 0.0.0.255
 rule 1 deny ip
#
ipsec policy testb 1 isakmp
 security acl 3001
 ike-profile aa
 transform-set 1
```

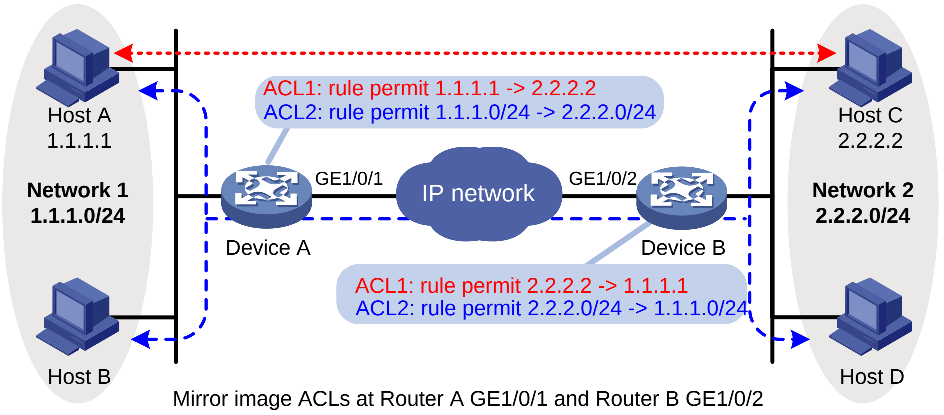
On Device A, apply the IPsec policy **testa** to the outbound interface of Device A. The IPsec policy contains two policy entries, **testa 1** and **testa 2**. The ACLs used by the two policy entries each contain a rule that matches traffic from 1.1.2.0/24 to 3.3.3.0/24. The one used in the policy entry **testa 1** is a deny statement and the one used in the policy entry **testa 2** is a permit statement. Because **testa 1** is matched prior to **testa 2**, traffic from 1.1.2.0/24 to 3.3.3.0/24 will match the deny statement and be sent as normal traffic. When the traffic arrives at Device B, the traffic matches rule 0 (a permit statement) in ACL 3001 used in the applied IPsec policy **testb**. Because non-IPsec traffic that matches a permit statement must be dropped on the inbound interface, Device B drops the traffic.

To make sure subnet 1.1.2.0/24 can access subnet 3.3.3.0/24, you can delete the deny rule in ACL 3000 on Device A.

Mirror image ACLs

To make sure SAs can be set up and the traffic protected by IPsec can be processed correctly between two IPsec peers, create mirror image ACLs on the IPsec peers. As shown in Figure 7, ACL rules on Device B are mirror images of the rules on Device A. In this way, SAs can be created successfully for the traffic between Host A and Host C and for the traffic between Network 1 and Network 2.

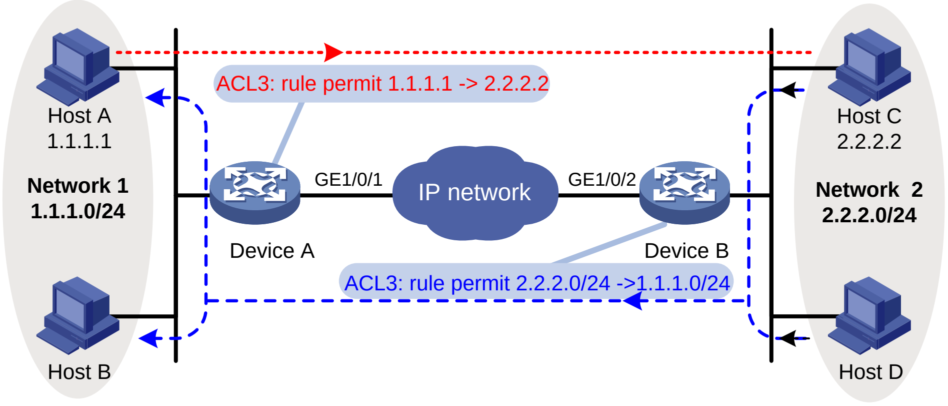
Figure 7 Mirror image ACLs


Host A
1.1.1.1
Network 1
1.1.1.0/24
Host B
ACL1: rule permit 1.1.1.1 -> 2.2.2.2
ACL2: rule permit 1.1.1.0/24 -> 2.2.2.0/24
GE1/0/1
IP network
GE1/0/2
Device A
Device B
ACL1: rule permit 2.2.2.2 -> 1.1.1.1
ACL2: rule permit 2.2.2.0/24 -> 1.1.1.0/24
Host C
2.2.2.2
Network 2
2.2.2.0/24
Host D
Mirror image ACLs at Router A GE1/0/1 and Router B GE1/0/2

If the ACL rules on IPsec peers do not form mirror images of each other, SAs can be set up only when both of the following requirements are met:

- The range specified by an ACL rule on one peer is covered by its counterpart ACL rule on the other peer. As shown in Figure 8, the range specified by the ACL rule configured on Device A is covered by its counterpart on Device B.
- The peer with the narrower rule initiates SA negotiation. If a wider ACL rule is used by the SA initiator, the negotiation request might be rejected because the matching traffic is beyond the scope of the responder. As shown in Figure 8, the SA negotiation initiated by Host A to Host C is accepted but the SA negotiations from Host C to Host A, from Host C to Host B, and from Host D to Host A are rejected.

Figure 8 Non-mirror image ACLs



ACL for IPsec protection in VPN instances

To use IPsec to protect the data of a VPN instance, you must specify the VPN instance for the protected data in the ACL.

As shown in Figure 9, to protect traffic of VPN1 by using IPsec, you must configure the ACL on Device A as follows:

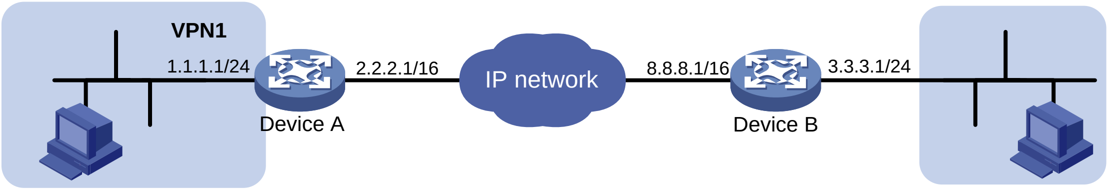
```
#
acl advanced 3400
 rule 0 permit ip vpn-instance vpn1 source 1.1.1.0 0.0.0.255 destination 3.3.3.0 0.0.0.255
#
```

In addition, you must specify VPN1 as the inside VPN instance in the IKE profile.

```
#
```

```
ike profile vpn1
 keychain vpn1
 match remote identity address 8.8.8.1 255.255.255.255
 inside-vpn vpn-instance vpn1
#
```

Figure 9 IPsec for VPN instances



Configuring an IPsec transform set

About this task

An IPsec transform set, part of an IPsec policy, defines the security parameters for IPsec SA negotiation, including the security protocol, encryption algorithms, and authentication algorithms.

Restrictions and guidelines

Changes to an IPsec transform set affect only SAs negotiated after the changes. To apply the changes to existing SAs, execute the **reset ipsec sa** command to clear the SAs so that they can be set up by using the updated parameters.

The transport mode applies only when the source and destination IP addresses of data flows match those of the IPsec tunnel. IPsec transform sets used in IPsec profiles for IPv6 routing protocols support only the transport mode.

The tunnel mode typically applies when the source and destination IP addresses of data flows are different from those of the IPsec tunnel. IPsec transform sets used in IPsec profiles for tunnels support only the tunnel mode.

When you configure the Perfect Forward Secrecy (PFS) feature in an IPsec transform set, follow these guidelines:

- In IKEv1, the security level of the DH group of the initiator must be higher than or equal to that of the responder. This restriction does not apply to IKEv2.
- The end without the PFS feature performs SA negotiation according to the PFS requirements of the peer end.

You can specify multiple authentication or encryption algorithms for the same security protocol. The algorithm specified earlier has a higher priority.

Some algorithms are available only for IKEv2. See Table 1.

Table 1 Algorithms available only for IKEv2

Type	Algorithms
Encryption algorithm	aes-ctr-128 aes-ctr-192 aes-ctr-256 camellia-cbc-128 camellia-cbc-192

	camellia-cbc-256 gmac-128 gmac-192 gmac-256 gcm-128 gcm-192 gcm-256
Authentication algorithm	aes-xcbc-mac
PFS algorithm	dh-group19 dh-group20

Procedure

1. Enter system view.

   system-view

2. Create an IPsec transform set and enter its view.

   ipsec transform-set *transform-set-name*

3. Specify the security protocol for the IPsec transform set.

   protocol { **ah** | **ah-esp** | **esp** }

   By default, the ESP security protocol is used.

4. Specify the encryption algorithms for ESP. Skip this step if the **protocol ah** command is configured.

   esp encryption-algorithm { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256** | **camellia-cbc-128** | **camellia-cbc-192** | **camellia-cbc-256** | **des-cbc** | **gmac-128** | **gmac-192** | **gmac-256** | **gcm-128** | **gcm-192** | **gcm-256** | **null** | **sm1-cbc-128** | **sm4-cbc** } *

   By default, no encryption algorithm is specified for ESP.

5. Specify the authentication algorithms for ESP. Skip this step if the **protocol ah** command is configured.

   esp authentication-algorithm { **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512** | **sm3** } *

   By default, no authentication algorithm is specified for ESP.

   The **aes-xcbc-mac** algorithm is available only for IKEv2.

6. Specify the authentication algorithms for AH. Skip this step if the **protocol esp** command is configured.

   ah authentication-algorithm { **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512** | **sm3** } *

   By default, no authentication algorithm is specified for AH.

   The **aes-xcbc-mac** algorithm is available only for IKEv2.

7. Specify the packet encapsulation mode.

   encapsulation-mode { **transport** | **tunnel** }

   By default, the security protocol encapsulates IP packets in tunnel mode.

8. (Optional.) Enable the PFS feature.

   pfs { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** | **dh-group24** | **dh-group19** | **dh-group20** }

   By default, the PFS feature is disabled.

For more information about PFS, see "Configuring IKE."

9. (Optional.) Enable the Extended Sequence Number (ESN) feature.

   esn enable [**both**]

   By default, the ESN feature is disabled.

   The ESN feature applies only to IPsec SAs negotiated by IKEv2.

Configuring a manual IPsec policy

In a manual IPsec policy, the parameters are configured manually, such as the keys, the SPIs, and the IP addresses of the two ends in tunnel mode.

Restrictions and guidelines

When you configure a manual IPsec policy, make sure the IPsec configuration at both ends of the IPsec tunnel meets the following requirements:

- The IPsec policies at the two ends must have IPsec transform sets that use the same security protocols, security algorithms, and encapsulation mode.
- The remote IPv4 address configured on the local end must be the same as the primary IPv4 address of the interface applied with the IPsec policy at the remote end. The remote IPv6 address configured on the local end must be the same as the first IPv6 address of the interface applied with the IPsec policy at the remote end.
- At each end, configure parameters for both the inbound SA and the outbound SA, and make sure the SAs in each direction are unique: For an outbound SA, make sure its triplet (remote IP address, security protocol, and SPI) is unique. For an inbound SA, make sure its SPI is unique.
- The local inbound SA must use the same SPI and keys as the remote outbound SA. The same is true of the local outbound SA and remote inbound SA.
- The keys for the IPsec SAs at the two tunnel ends must be configured in the same format. For example, if the local end uses a key in hexadecimal format, the remote end must also use a key in hexadecimal format. If you configure a key in both the character and the hexadecimal formats, only the most recent configuration takes effect.
- If you configure a key in character format for ESP, the device automatically generates an authentication key and an encryption key for ESP.

Procedure

1. Enter system view.

   system-view

2. Create a manual IPsec policy entry and enter its view.

   ipsec { **ipv6-policy** | **policy** } *policy-name seq-number* **manual**

3. (Optional.) Configure a description for the IPsec policy.

   description *text*

   By default, no description is configured.

4. Specify an ACL for the IPsec policy.

   security acl [**ipv6**] { *acl-number* | **name** *acl-name* }

   By default, no ACL is specified for an IPsec policy.

   You can specify only one ACL for an IPsec policy.

5. Specify an IPsec transform set for the IPsec policy.

   transform-set *transform-set-name*

   By default, no IPsec transform set is specified for an IPsec policy.

   You can specify only one IPsec transform set for a manual IPsec policy.

6. Specify the remote IP address of the IPsec tunnel.

   remote-address { *ipv4-address* | **ipv6** *ipv6-address* }

   By default, the remote IP address of the IPsec tunnel is not specified.

7. Configure an SPI for the inbound IPsec SA.

   sa spi inbound { **ah** | **esp** } *spi-number*

   By default, no SPI is configured for the inbound IPsec SA.

8. Configure an SPI for the outbound IPsec SA.

   sa spi outbound { **ah** | **esp** } *spi-number*

   By default, no SPI is configured for the outbound IPsec SA.

9. Configure keys for the IPsec SA.

   - Configure an authentication key in hexadecimal format for AH.

     sa hex-key authentication { **inbound** | **outbound** } **ah** { **cipher** | **simple** } *string*

   - Configure an authentication key in character format for AH.

     sa string-key { **inbound** | **outbound** } **ah** { **cipher** | **simple** } *string*

   - Configure a key in character format for ESP.

     sa string-key { **inbound** | **outbound** } **esp** { **cipher** | **simple** } *string*

   - Configure an authentication key in hexadecimal format for ESP.

     sa hex-key authentication { **inbound** | **outbound** } **esp** { **cipher** | **simple** }

   - Configure an encryption key in hexadecimal format for ESP.

     sa hex-key encryption { **inbound** | **outbound** } **esp** { **cipher** | **simple** } *string*

   By default, no keys are configured for the IPsec SA.

   Configure keys correctly for the security protocol (AH, ESP, or both) you have specified in the IPsec transform set used by the IPsec policy.

Configuring an IKE-based IPsec policy

About this task

In an IKE-based IPsec policy, the parameters are automatically negotiated through IKE.

To configure an IKE-based IPsec policy, use one of the following methods:

- Directly configure it by configuring the parameters in IPsec policy view.
- Configure it by using an existing IPsec policy template with the parameters to be negotiated configured.

  A device using an IPsec policy that is configured in this way cannot initiate an SA negotiation, but it can respond to a negotiation request. The parameters not defined in the template are determined by the initiator. For example, in an IPsec policy template, the ACL is optional. If you do not specify an ACL, the IPsec protection range has no limit. So the device accepts all ACL settings of the negotiation initiator.

  When the remote end's information (such as the IP address) is unknown, this method allows the remote end to initiate negotiations with the local end.

The configurable parameters for an IPsec policy template are the same as those when you directly configure an IKE-based IPsec policy. The difference is that more parameters are optional for an IPsec policy template. Except the IPsec transform sets and the IKE profile, all other parameters are optional.

Restrictions and guidelines for IKE-based IPsec policy configuration

The IPsec policies at the two tunnel ends must have IPsec transform sets that use the same security protocols, security algorithms, and encapsulation mode.

The IPsec policies at the two tunnel ends must have the same IKE profile parameters.

An IKE-based IPsec policy can use a maximum of six IPsec transform sets. During an IKE negotiation, IKE searches for a fully matched IPsec transform set at the two ends of the IPsec tunnel. If no match is found, no SA can be set up, and the packets expecting to be protected will be dropped.

The remote IP address of the IPsec tunnel is required on an IKE negotiation initiator and is optional on the responder. The remote IP address specified on the local end must be the same as the local IP address specified on the remote end.

The IPsec SA uses the local lifetime settings or those proposed by the peer, whichever are smaller.

The IPsec SA can have both a time-based lifetime and a traffic-based lifetime. The IPsec SA expires when either lifetime expires.

If you specify both an IKEv1 profile and an IKEv2 profile for an IPsec policy, the IKEv2 profile is used preferentially. For more information about IKEv1 and IKEv2 profiles, see "Configuring IKE" and "Configuring IKEv2."

Directly configuring an IKE-based IPsec policy

1. Enter system view.

   system-view

2. Create an IKE-based IPsec policy entry and enter its view.

   ipsec { **ipv6-policy** | **policy** } *policy-name seq-number* **isakmp**

3. (Optional.) Configure a description for the IPsec policy.

   description *text*

   By default, no description is configured.

4. (Optional.) Set the IPsec SA negotiation triggering mode.

   sa trigger-mode { **auto** | **traffic-based** }

   By default, IPsec SA negotiation is triggered when traffic requires IPsec protection.

   If the ACL for an IPsec policy or IPsec policy template uses the aggregation or the per-host mode, the IPsec policy or IPsec policy template cannot trigger IPsec SA negotiation in auto mode.

5. Specify an ACL for the IPsec policy.

   security acl [**ipv6**] { *acl-number* | **name** *acl-name* } [**aggregation** | **per-host**]

   By default, no ACL is specified for an IPsec policy.

   You can specify only one ACL for an IPsec policy.

6. Specify IPsec transform sets for the IPsec policy.

   transform-set *transform-set-name*&<1-6>

   By default, no IPsec transform sets are specified for an IPsec policy.

7. Specify an IKE profile or IKEv2 profile for the IPsec policy.
   - Specify an IKE profile.

     ike-profile *profile-name*

     By default, no IKE profile is specified for an IPsec policy.
   - Specify an IKEv2 profile.

     ikev2-profile *profile-name*

     By default, no IKEv2 profile is specified for an IPsec policy.

8. Specify the local IP address of the IPsec tunnel.

   local-address { *ipv4-address* | **ipv6** *ipv6-address* }

   By default, the local IPv4 address of the IPsec tunnel is the primary IPv4 address of the interface to which the IPsec policy is applied. The local IPv6 address of the IPsec tunnel is the first IPv6 address of the interface to which the IPsec policy is applied.

   The local IP address specified by this command must be the same as the IP address used as the local IKE identity.

   The local address cannot be a secondary IP address of the interface where the IPsec policy is applied.

9. Specify the remote IP address of the IPsec tunnel.

   remote-address { [**ipv6**] *host-name* | *ipv4-address* | **ipv6** *ipv6-address* } [**primary**]

   By default, the remote IP address of the IPsec tunnel is not specified.

10. (Optional.) Set the lifetime, soft lifetime buffer, or idle timeout for the IPsec SA.

    - Set the IPsec SA lifetime.

      sa duration { **time-based** *seconds* | **traffic-based** *kilobytes* }

      By default, the global SA lifetime is used.

    - Set the time-based or traffic-based IPsec SA soft lifetime buffer.

      sa soft-duration buffer { **time-based** *seconds* | **traffic-based** *kilobytes* }

      By default, no IPsec SA soft lifetime buffers are configured.

    - Set the IPsec SA idle timeout.

      sa idle-time *seconds*

      By default, the global IPsec SA idle timeout is used.

11. (Optional.) Enable the Traffic Flow Confidentiality (TFC) padding feature.

    tfc enable

    By default, the TFC padding feature is disabled.

    TFC padding applies only to IPsec SAs negotiated by IKEv2.

Configuring an IKE-based IPsec policy by using an IPsec policy template

1. Enter system view.

   system-view

2. Create an IPsec policy template and enter its view.

   ipsec { **ipv6-policy-template** | **policy-template** } *template-name* *seq-number*

3. (Optional.) Configure a description for the IPsec policy template.

   description *text*

   By default, no description is configured.

4. (Optional.) Specify an ACL for the IPsec policy template.

   security acl [**ipv6**] { *acl-number* | **name** *acl-name* } [**aggregation** | **per-host**]

   By default, no ACL is specified for an IPsec policy template.

   You can specify only one ACL for an IPsec policy template.

5. Specify IPsec transform sets for the IPsec policy template.

   transform-set *transform-set-name*&<1-6>

   By default, no IPsec transform sets are specified for an IPsec policy template.

6. Specify an IKE profile or IKEv2 profile for the IPsec policy template.
   - Specify an IKE profile.

     ike-profile *profile-name*

     By default, no IKE profile is specified for an IPsec policy template.

     Make sure the specified IKE profile is not used by another IPsec policy or IPsec policy template.
   - Specify an IKEv2 profile.

     ikev2-profile *profile-name*

     By default, no IKEv2 profile is specified for an IPsec policy template.
7. Specify the local IP address of the IPsec tunnel.

   local-address { *ipv4-address* | **ipv6** *ipv6-address* }

   The default local IPv4 address and IPv6 address is the primary IPv4 address and first IPv6 address of the interface where the IPsec policy is applied.

   The local IP address specified by this command must be the same as the IP address used as the local IKE identity.

   The local address cannot be a secondary IP address of the interface where the IPsec policy is applied.
8. Specify the remote IP address of the IPsec tunnel.

   remote-address { [**ipv6**] *host-name* | *ipv4-address* | **ipv6** *ipv6-address* }

   By default, the remote IP address of the IPsec tunnel is not specified.
9. (Optional.) Set the lifetime and idle timeout for the IPsec SA.
   - Set the IPsec SA lifetime.

     sa duration { **time-based** *seconds* | **traffic-based** *kilobytes* }

     By default, the global SA lifetime is used.
   - Set the IPsec SA idle timeout.

     sa idle-time *seconds*

     By default, the global IPsec SA idle timeout is used.
10. (Optional.) Enable the Traffic Flow Confidentiality (TFC) padding feature.

    tfc enable

    By default, the TFC padding feature is disabled.
11. Return to system view.

    quit
12. Create an IKE-based IPsec policy by using the IPsec policy template.

    ipsec { **ipv6-policy** | **policy** } *policy-name seq-number* **isakmp template** *template-name*

Applying an IPsec policy to an interface

Restrictions and guidelines

An IKE-based IPsec policy that is bound to a source interface can be applied to multiple interfaces.

A manual IPsec policy can be applied to only one interface.

To cancel the IPsec protection, remove the application of the IPsec policy.

Procedure

1. Enter system view.

system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Apply an IPsec policy to the interface.

   ipsec apply { **ipv6-policy** | **policy** } *policy-name*

   By default, no IPsec policy is applied to an interface.

   On one interface, you can apply a maximum of two IPsec policies: one IPv4 IPsec policy and one IPv6 IPsec policy.

Enabling ACL checking for de-encapsulated packets

About this task

This feature compares the de-encapsulated incoming IPsec packets against the ACL in the IPsec policy and discards those that do not match any permit rule of the ACL. This feature can protect networks against attacks using forged IPsec packets.

This feature applies only to tunnel-mode IPsec.

Procedure

1. Enter system view.

   system-view

2. Enable ACL checking for de-encapsulated packets.

   ipsec decrypt-check enable

   By default, ACL checking for de-encapsulated packets is enabled.

Configuring IPsec anti-replay

About this task

IPsec anti-replay protects networks against anti-replay attacks by using a sliding window mechanism called anti-replay window. This feature checks the sequence number of each received IPsec packet against the current IPsec packet sequence number range of the sliding window. If the sequence number is not in the current sequence number range, the packet is considered a replayed packet and is discarded.

IPsec packet de-encapsulation involves complicated calculation. De-encapsulation of replayed packets is not required, and the de-encapsulation process consumes large amounts of resources and degrades performance, resulting in DoS. IPsec anti-replay can check and discard replayed packets before de-encapsulation.

In some situations, service data packets are received in a different order than their original order. The IPsec anti-replay feature drops them as replayed packets, which impacts communications. If this happens, disable IPsec anti-replay checking or adjust the size of the anti-replay window as required.

Restrictions and guidelines

IPsec anti-replay does not affect manually created IPsec SAs. According to the IPsec protocol, only IKE-based IPsec SAs support anti-replay.

Set the anti-replay window size as small as possible to reduce the impact on system performance.

Failure to detect anti-replay attacks might result in denial of services. If you want to disable IPsec anti-replay, make sure you understand the impact of the operation on network security.

Procedure

1. Enter system view.

   system-view

2. Enable IPsec anti-replay.

   ipsec anti-replay check

   By default, IPsec anti-replay is enabled.

3. Set the size of the IPsec anti-replay window.

   ipsec anti-replay window *width*

   The default size is 64.

Binding a source interface to an IPsec policy

About this task

For high availability, a core device is usually connected to an ISP through two links, which operate in backup or load sharing mode. The two interfaces negotiate with their peers to establish IPsec SAs respectively. When one interface fails and a link failover occurs, the other interface needs to take some time to renegotiate SAs, resulting in service interruption.

To solve these problems, bind a source interface to an IPsec policy and apply the policy to both interfaces. This enables the two physical interfaces to use the same source interface to negotiate IPsec SAs. As long as the source interface is up, the negotiated IPsec SAs will not be removed and will keep working, regardless of link failover.

Restrictions and guidelines

Only the IKE-based IPsec policies can be bound to a source interface.

An IPsec policy can be bound to only one source interface.

A source interface can be bound to multiple IPsec policies.

If the source interface bound to an IPsec policy is removed, the IPsec policy becomes a common IPsec policy.

If no local address is specified for an IPsec policy that has been bound to a source interface, the IPsec policy uses the IP address of the bound source interface to perform IKE negotiation. If a local address is specified, the IPsec policy uses the local address to perform IKE negotiation.

Procedure

1. Enter system view.

   system-view

2. Bind a source interface to an IPsec policy.

   ipsec { **ipv6-policy** | **policy** } *policy-name* **local-address** *interface-type interface-number*

   By default, no source interface is bound to an IPsec policy.

Enabling QoS pre-classify

About this task

When both an IPsec policy and a QoS policy are applied to an interface, QoS classifies packets by using the new headers added by IPsec. If you want QoS to classify packets by using the headers of the original IP packets, enable the QoS pre-classify feature.

Restrictions and guidelines

If you configure both IPsec and QoS on an interface, make sure the IPsec traffic classification rules match the QoS traffic classification rules. If the rules do not match, QoS might classify the packets of one IPsec SA to different queues, causing packets to be sent out of order. When IPsec anti-replay is enabled, IPsec will drop the incoming packets that are out of the anti-replay window, resulting in packet loss.

IPsec traffic classification rules are determined by the rules of the specified ACL. For more information about QoS policy and classification, see *ACL and QoS Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPsec policy view or IPsec policy template view.
   - Enter IPsec policy view.

     `ipsec` { **`ipv6-policy`** | **`policy`** } *`policy-name`* *`seq-number`* [**`isakmp`** | **`manual`**]
   - Enter IPsec policy template view.

     `ipsec` { **`ipv6-policy-template`** | **`policy-template`** } *`template-name`* *`seq-number`*

3. Enable QoS pre-classify.

   `qos pre-classify`

   By default, QoS pre-classify is disabled.

Configuring IPsec RRI

Restrictions and guidelines

Enabling IPsec RRI for an IPsec policy deletes all existing IPsec SAs created by this IPsec policy. IPsec RRI creates static routes according to new IPsec SAs.

Disabling IPsec RRI for an IPsec policy deletes all existing IPsec SAs created by this IPsec policy and the associated static routes.

IPsec RRI is supported in both tunnel mode and transport mode.

If you change the preference value or tag value for an IPsec policy, the device deletes all IPsec SAs created by this IPsec policy, and the associated static routes. The change takes effect for future IPsec RRI-created static routes.

IPsec RRI does not generate a static route to a destination address to be protected if the destination address is not defined in the ACL used by an IPsec policy or an IPsec policy template. You must manually configure a static route to the destination address.

Procedure

1. Enter system view.

   `system-view`

2. Enter IPsec policy view or IPsec policy template view.
   - Enter IPsec policy view.

     `ipsec` { **`policy`** | **`ipv6-policy`** } *`policy-name`* *`seq-number`* **`isakmp`**
   - Enter IPsec policy template view.

     `ipsec` { **`ipv6-policy-template`** | **`policy-template`** } *`template-name`* *`seq-number`*

3. Enable IPsec RRI.

reverse-route [**next-hop** [**ipv6**] *ip-address*] **dynamic**

By default, IPsec RRI is disabled.

4. (Optional.) Set the preference value for the static routes created by IPsec RRI.

   reverse-route preference *number*

   The default value is 60.
5. (Optional.) Set the tag value for the static routes created by IPsec RRI.

   reverse-route tag *tag-value*

   The default value is 0.

Configuring IPsec smart link selection

About this task

IPsec smart link selection enables a branch gateway to dynamically select a qualified link among the available links to establish the IPsec tunnel to the headquarters.

To configure IPsec smart link selection on a branch gateway, perform the following tasks:

1. Configure an IPsec smart link policy.

   An IPsec smart link policy defines the following settings:
   - Link quality probe settings, such as the number of probe packets sent in each probe cycle and the probe packet sending interval.
   - Links available for smart link selection. A link configured earlier has a higher priority in smart link selection.
   - Link switchover thresholds, including the packet loss ratio threshold and delay threshold.
2. Apply the IPsec smart link policy to an IKE-based IPsec policy.

The IPsec smart link selection process is as follows:

1. When the branch gateway identifies traffic that needs IPsec protection for the first time, it establishes an IPsec tunnel to the headquarters over the link with the highest priority.

   After a link is selected for IPsec tunnel establishment, the device applies the IPsec policy that uses the IPsec smart link policy to the local interface of the link.
2. The device periodically sends probe packets to test the packet loss ratio and delay over the active link.
3. If the packet loss ratio or delay over the link exceeds the configured threshold, the device starts a cyclic link switchover process, during which a qualified link is selected to transfer traffic.

   Cyclic link switchover probes the available links one by one in descending order of the link priority, and uses the first qualified link to transfer traffic. If no links are qualified when the maximum number of link switchover cycles is reached, the device selects a link for traffic as follows:
   - Selects the link with the lowest packet loss ratio.
   - Selects the link with the lowest delay if the links have the same packet loss ratio.
   - Selects the link with the lowest priority if the links have the same packet loss ratio and delay.

   After 10 minutes, the device starts link quality probing and cyclic link switchovers again.

Restrictions and guidelines

IPsec smart link selection is supported only on IPv4 networks.

An IPsec smart link policy takes effect after it is applied to an IKE-based IPsec policy. You cannot apply an IPsec smart link policy to a manual IPsec policy or an IPsec policy created by using an IPsec policy template.

When you configure IPsec smart link selection on the IPsec gateway at a branch, follow these restrictions and guidelines:

- An IPsec smart link policy can be applied to only one IPsec policy, and an IPsec policy can use only one IPsec smart link policy.
- In the IPsec policy that uses an IPsec smart link policy, do not use **`local-address`** and
- Make sure the device does not have static routes that use the same destination addresses as the routes automatically generated by smart link selection. If such static routes exist, delete them.

When you configure the IPsec gateway at the headquarters, follow these restrictions and guidelines:

- The IPsec policy applied to the gateway interface used for communication with the branch gateway must be created by using an IPsec policy template.
- When you specify the remote IP address in IKE configuration, use 0.0.0.0 0 or specify all the remote IP addresses configured at the branch gateway.

Procedure

1. Enter system view.

   `system-view`

2. Configure an IPsec smart link policy.

   a. Create an IPsec smart link policy and enter its view.

   `ipsec smart-link policy` *`policy-name`*

   b. Configure links for IPsec smart link selection.

   `link` *`link-id`* **`interface`** *`interface-type interface-number`* [**`local`** *`local-address`* **`nexthop`** *`nexthop-address`*] **`remote`** *`remote-address`*

   By default, an IPsec smart link policy does not contain any links.

   c. Enable IPsec smart link selection.

   `smart-link enable`

   By default, IPsec smart link selection is disabled in an IPsec smart link policy.

   The device performs link quality probing and cyclic link switchovers only when IPsec smart link selection is enabled in the IPsec smart link policy.

   d. (Optional.) Manually activate a link.

   `activate link` *`link-id`*

   To establish an IPsec tunnel over a specific link, use this command to activate the link.

   If the packet loss ratio or delay over the link exceeds the configured threshold, the first link switchover cycle starts from the manually activated link and ends with the link that has the lowest priority.

   e. (Optional.) Move links to adjust their priorities.

   `move link` *`link-id1`* **`before`** *`link-id2`*

   By default, a link configured earlier has a higher priority in smart link selection.

   During link switchover, traffic will be switched over the links in descending order of the link priority.

   f. (Optional.) Set the maximum number of link switchover cycles.

   `link-switch cycles` *`number`*

   By default, the maximum number of link switchover cycles is 3.

   If you set *`number`* to 0, the device never stops link quality probing and cyclic link switchovers.

   g. Set the number of link quality probe packets sent in each probe cycle and the probe packet sending interval.

link-probe { **interval** *interval* | **count** *number* }

By default, the device sends 10 probe packets at 1-second intervals in each probe cycle.

h. Specify the source and destination IP addresses for the link quality probe packets.

link-probe source *source-address* **destination** *destination-address*

By default, the link quality probe packets use the local and remote IP addresses of the probed link as the source and destination IP addresses.

i. Set the link switchover thresholds.

link-switch threshold { **loss** *loss-ratio* | **delay** *delay* }

By default, the packet loss ratio threshold is 30%, and the delay threshold is 500 milliseconds.

3. Configure the gateway address for the local interface of a link.

a. Return to system view.

quit

b. Enter interface view.

interface *interface-type* { *interface-number* | *interface-number.subnumber* }

c. Specify the gateway address for the interface.

gateway *gateway-address* [**no-route**]

By default, an interface with a manually configured IP address does not have a gateway address.

The local interface of a link is the interface specified when you configure the link by using the **link** command.

This task is required for the local interface of each link that meets the following conditions:

- The IP address of the interface is manually configured.
- The **nexthop** *nexthop-address* option is not specified in the **link** command.

The **gateway** command does not take effect if the interface acquires its IP address through DHCP or PPPoE. Such interfaces always use the gateway address assigned by the DHCP or PPPoE server.

4. Apply the IPsec smart link policy to an IKE-based IPsec policy

a. Return to system view.

quit

b. Enter the view of an IKE-based IPsec policy.

ipsec policy *policy-name seq-number* **isakmp**

c. Apply the IPsec smart link policy to the IPsec policy.

smart-link policy *policy-name*

By default, an IPsec smart link policy is not applied to an IPsec policy.

Configuring IPsec netmask filtering

About this task

On a hub-spoke network, if the IPsec data flow range configured on a spoke is too large, traffic of other spokes might be directed to that spoke incorrectly. To avoid incorrect packet forwarding, you can enable IPsec netmask filtering on the hub device. When negotiating an IPsec SA for a data flow, the device checks the mask lengths of the data flow. The IPsec SA negotiation proceeds only if the mask lengths of the source and destination IP addresses of the data flow are greater than or equal to those configured by IPsec netmask filtering. If the data flow fails to pass the netmask filtering, the IPsec SA negotiation fails and the device generates a corresponding SA negotiation failure notification. On receiving such notifications, you must reconfigure the ACL settings for IPsec on the spoke devices.

Restrictions and guidelines

This feature is supported only on IPv4 networks.

This feature takes effect only IPsec SAs negotiated by using IPsec policy templates.

As a best practice, configure this feature on the hub device of a hub-spoke network.

Procedure

1. Enter system view.

   `system-view`

2. Configure IPsec netmask filtering.

   `ipsec netmask-filter` { **`destination-mask`** *`mask-length`* | **`source-mask`** *`mask-length`* } *

   By default, IPsec netmask filtering is not configured.

Enabling IPsec flow overlap check

About this task

In a hub-spoke network, the hub typically uses an IPsec policy template to negotiate IPsec SAs with spokes. The data flows to be protected by the IPsec SAs might overlap with each other. To avoid IPsec flow overlapping, you can enable IPsec flow overlap check on the hub device. When negotiating an IPsec SA for a data flow, the device checks whether the data flow overlaps with an existing protected data flow. If yes, the new IPsec SA negotiation fails and the device generates an IPsec flow overlap notification. On receiving such notifications, you must reconfigure the ACL settings for IPsec on the spoke devices.

This feature checks the destination IP address of a data flow to be protected with the destination IP addresses of the existing protected data flows. If an overlap exists, the feature determines that the data flow overlaps.

Restrictions and guidelines

The following applies to the IPsec flow overlap check feature:

- As a best practice, enable this feature on the hub device of a hub-spoke network.
- This feature takes effect only on IPsec SAs negotiated by using IPsec policy templates.
- This feature takes effect only for new IPsec SA negotiations. It does not take effect on existing IPsec SAs.
- This feature takes effect only on IPsec SAs negotiated on the same interface and in the same VPN instance.
- This feature does not take effect on renegotiated IPsec SAs.
- This feature does not check overlaps for source IP addresses of data flows.
- This feature impacts device performance, enable this feature only when necessary (for example, for network upgrade or expansion) and disable it in time.

Procedure

1. Enter system view.

   `system-view`

2. Enable IPsec flow overlap check.

   `ipsec flow-overlap check enable`

   By default, IPsec flow overlap check is disabled.

Configuring IPsec for IPv6 routing protocols

IPsec protection for IPv6 routing protocols tasks at a glance

To configure IPsec protection for IPv6 routing protocols, perform the following tasks:

1. Configuring an IPsec transform set
2. Configuring a manual IPsec profile
3. Applying the IPsec profile to an IPv6 routing protocol
4. (Optional.) Configuring IPsec anti-replay redundancy
5. (Optional.) Configuring IPsec fragmentation
6. (Optional.) Setting the maximum number of IPsec tunnels
7. (Optional.) Enabling logging for IPsec packets
8. (Optional.) Enabling logging for IPsec negotiation
9. (Optional.) Configuring SNMP notifications for IPsec

Configuring a manual IPsec profile

About this task

A manual IPsec profile specifies the IPsec transform set used for protecting data flows, and the SPIs and keys used by the SAs.

Restrictions and guidelines

When you configure a manual IPsec profile, make sure the IPsec profile configuration at both tunnel ends meets the following requirements:

- The IPsec transform set specified in the IPsec profile at the two tunnel ends must have the same security protocol, encryption and authentication algorithms, and packet encapsulation mode.
- The local inbound and outbound IPsec SAs must have the same SPI and key.
- The IPsec SAs on the devices in the same scope must have the same key. The scope is defined by protocols. For OSPFv3, the scope consists of OSPFv3 neighbors or an OSPFv3 area. For RIPng, the scope consists of directly-connected neighbors or a RIPng process. For BGP, the scope consists of BGP peers or a BGP peer group.
- The keys for the IPsec SAs at the two tunnel ends must be configured in the same format. For example, if the local end uses a key in hexadecimal format, the remote end must also use a key in hexadecimal format. If you configure a key in both the character and the hexadecimal formats, only the most recent configuration takes effect.
- If you configure a key in character format for ESP, the device automatically generates an authentication key and an encryption key for ESP.

Procedure

1. Enter system view.

   system-view

2. Create a manual IPsec profile and enter its view.

   ipsec profile *profile-name* **manual**

   The **manual** keyword is not needed if you enter the view of an existing IPsec profile.

3. (Optional.) Configure a description for the IPsec profile.

   description *text*

By default, no description is configured.

4. Specify an IPsec transform set.

   `transform-set` *`transform-set-name`*

   By default, no IPsec transform set is specified in an IPsec profile.

   The specified IPsec transform set must use the transport mode.

5. Configure an SPI for an SA.

   `sa spi` { **`inbound`** | **`outbound`** } { **`ah`** | **`esp`** } *`spi-number`*

   By default, no SPI is configured for an SA.

6. Configure keys for the IPsec SA.

   - Configure an authentication key in hexadecimal format for AH.

     `sa hex-key authentication` { **`inbound`** | **`outbound`** } **`ah`** { **`cipher`** | **`simple`** } *`string`*

   - Configure an authentication key in character format for AH.

     `sa string-key` { **`inbound`** | **`outbound`** } **`ah`** { **`cipher`** | **`simple`** } *`string`*

   - Configure a key in character format for ESP.

     `sa string-key` { **`inbound`** | **`outbound`** } **`esp`** { **`cipher`** | **`simple`** } *`string`*

   - Configure an authentication key in hexadecimal format for ESP.

     `sa hex-key authentication` { **`inbound`** | **`outbound`** } **`esp`** { **`cipher`** | **`simple`** }

   - Configure an encryption key in hexadecimal format for ESP.

     `sa hex-key encryption` { **`inbound`** | **`outbound`** } **`esp`** { **`cipher`** | **`simple`** } *`string`*

   By default, no keys are configured for the IPsec SA.

   Configure a key for the security protocol (AH, ESP, or both) you have specified.

Applying the IPsec profile to an IPv6 routing protocol

For information about the configuration procedure, see IPv6 BGP, OSPFv3, and RIPng configuration in *Layer 3—IP Routing Configuration Guide*.

Configuring IPsec for tunnel interfaces

IPsec protection for tunnel interfaces tasks at a glance

To configure IPsec protection for tunnel interfaces, perform the following tasks:

1. Configuring an IPsec transform set
2. Configuring an IKE-based IPsec profile
3. Applying an IKE-based IPsec profile to a tunnel interface
4. (Optional.) Configuring IPsec anti-replay redundancy
5. (Optional.) Configuring the global IPsec SA lifetime and idle timeout
6. (Optional.) Configuring IPsec fragmentation
7. (Optional.) Setting the maximum number of IPsec tunnels
8. (Optional.) Enabling logging for IPsec packets
9. (Optional.) Enabling logging for IPsec negotiation
10. (Optional.) Configuring SNMP notifications for IPsec

Configuring an IKE-based IPsec profile

An IKE-based IPsec profile specifies the IPsec transform sets used for protecting data flows, and the IKE profile used for IKE negotiation.

Restrictions and guidelines

The IPsec profiles at the two tunnel ends must have IPsec transform sets that use the same security protocols, security algorithms, and encapsulation mode.

The IPsec profiles at the two tunnel ends must have the same IKE profile parameters.

An IKE-based IPsec profile can use a maximum of six IPsec transform sets. During an IKE negotiation, IKE searches for a fully matched IPsec transform set at the two ends of the IPsec tunnel. If no match is found, no SA can be set up, and the packets expecting to be protected will be dropped.

The IPsec SA uses the local lifetime settings or those proposed by the peer, whichever are smaller.

The IPsec SA can have both a time-based lifetime and a traffic-based lifetime. The IPsec SA expires when either lifetime expires.

Procedure

1. Enter system view.

   `system-view`

2. Create an IKE-based IPsec profile and enter its view.

   `ipsec profile` *`profile-name`* **`isakmp`**

   The **`isakmp`** keyword is not needed if you enter the view of an existing IPsec profile.

3. (Optional.) Configure a description for the IPsec profile.

   `description` *`text`*

   By default, no description is configured.

4. Specify IPsec transform sets.

   `transform-set` *`transform-set-name`*`&<1-6>`

   By default, no IPsec transform sets are specified in an IPsec profile.

   The specified IPsec transform sets must use the tunnel mode.

5. Specify an IKE profile.

   `ike-profile` *`profile-name`*

   By default, no IKE profile is specified for an IPsec profile, and the device selects an IKE profile configured in system view for negotiation. If no IKE profile is configured in system view, the globally configured IKE settings are used.

   You can specify only one IKE profile for an IPsec profile.

   For more information about IKE profiles, see "Configuring IKE."

6. (Optional.) Specify an IKEv2 profile.

   `ikev2-profile` *`profile-name`*

   By default, no IKEv2 profile is specified for an IPsec profile. If both an IKEv1 profile and an IKEv2 profile are specified for an IPsec profile, the IKEv2 profile is preferred.

   You can specify only one IKEv2 profile for an IPsec profile. For more information about IKEv2 profiles, see "Configuring IKEv2."

7. (Optional.) Set the IPsec SA lifetime.

   `sa duration` `{` **`time-based`** *`seconds`* `|` **`traffic-based`** *`kilobytes`* `}`

   By default, the global SA lifetime is used.

8. (Optional.) Set the time-based or traffic-based IPsec SA soft lifetime buffer.

sa soft-duration buffer { **time-based** *seconds* | **traffic-based** *kilobytes* }

By default, no IPsec SA soft lifetime buffers are configured.

9. (Optional.) Set the IPsec SA idle timeout.

   sa idle-time *seconds*

   By default, the global SA idle timeout is used.

Applying an IKE-based IPsec profile to a tunnel interface

About this task

After an IKE-based IPsec profile is applied to a tunnel interface, the peers negotiate an IPsec tunnel through IKE to protect data transmitted through the tunnel interface. The tunnel interface comes up after the IKE negotiation succeeds.

When you specify the IPsec profile to be applied to a tunnel interface, you can specify an ACL to filter the packets routed to the tunnel interface. Only the ACL-permitted packets can be protected by IPsec.

Procedure

1. Enter system view.

   system-view

2. Create a tunnel interface and enter its view.

   interface tunnel *number* **mode** { **advpn** { **gre** | **udp** } [**ipv6**] | **ipsec** [**ipv6**] }

3. Apply an IKE-based IPsec profile to the tunnel interface.

   tunnel protection ipsec profile *profile-name* [**acl** [**ipv6**] { *acl-number* | **name** *acl-name* }]

   By default, no IPsec profile is applied to a tunnel interface.

Configuring IPsec anti-replay redundancy

About this task

This feature synchronizes the following information from the active device to the standby device at configurable packet-based intervals:

- Lower bound values of the IPsec anti-replay window for inbound packets.
- IPsec anti-replay sequence numbers for outbound packets.

This feature, used together with IPsec redundancy, ensures uninterrupted IPsec traffic forwarding and anti-replay protection when the active device fails.

Procedure

1. Enter system view.

   system-view

2. Enable IPsec redundancy.

   ipsec redundancy enable

   By default, IPsec redundancy is disabled.

3. Enter IPsec profile view, IPsec policy view or IPsec policy template view.

   - Enter IPsec profile view.

     ipsec profile *profile-name* [**manual** | **isakmp**]

- Enter IPsec policy view.

  ipsec { **ipv6-policy** | **policy** } *policy-name seq-number* [**isakmp** | **manual**]

- Enter IPsec policy template view.

  ipsec { **ipv6-policy-template** | **policy-template** } *template-name seq-number*

4. Set the anti-replay window synchronization interval for inbound packets and the sequence number synchronization interval for outbound packets.

   redundancy replay-interval inbound *inbound-interval* **outbound** *outbound-interval*

   By default, the active device synchronizes the anti-replay window every time it receives 1000 packets and synchronizes the sequence number every time it sends 100000 packets.

Configuring the global IPsec SA lifetime and idle timeout

About this task

If the IPsec SA lifetime and idle timeout are not configured in an IPsec policy, IPsec policy template, or IPsec profile, the global settings are used.

When IKE negotiates IPsec SAs, it uses the local lifetime settings or those proposed by the peer, whichever are smaller.

An IPsec SA can have both a time-based lifetime and a traffic-based lifetime. The IPsec SA expires when either lifetime expires.

Procedure

1. Enter system view.

   system-view

2. Set the global IPsec SA lifetime, soft lifetime buffer, or idle timeout.

   - Set the global IPsec SA lifetime.

     ipsec sa global-duration { **time-based** *seconds* | **traffic-based** *kilobytes* }

     By default, the time-based SA lifetime is 3600 seconds, and the traffic-based SA lifetime is 1843200 kilobytes.

   - Set the global time-based or traffic-based IPsec SA soft lifetime buffer.

     ipsec sa global-soft-duration buffer { **time-based** *seconds* | **traffic-based** *kilobytes* }

     By default, no global IPsec SA soft lifetime buffers are configured.

   - Set the global SA idle timeout.

     ipsec sa idle-time *seconds*

     By default, the global IPsec SA idle timeout feature is disabled.

Configuring IPsec fragmentation

About this task

Perform this task to configure the device to fragment packets before or after IPsec encapsulation.

If you configure the device to fragment packets before IPsec encapsulation, the device predetermines the encapsulated packet size before the actual encapsulation. If the encapsulated packet size exceeds the MTU of the output interface, the device fragments the packets before encapsulation. If a packet's DF bit is set, the device drops the packet and sends an ICMP error message.

If you configure the device to fragment packets after IPsec encapsulation, the device directly encapsulates the packets and fragments the encapsulated packets in subsequent service modules.

Restrictions and guidelines

This feature takes effect on IPsec protected IPv4 packets.

Procedure

1. Enter system view.
   system-view
2. Configure IPsec fragmentation.
   ipsec fragmentation { **after-encryption** | **before-encryption** }
   By default, the device fragments packets before IPsec encapsulation.

Configuring the DF bit of IPsec packets

About this task

Perform this task to configure the Don't Fragment (DF) bit in the new IP header of IPsec packets in one of the following ways:

- **clear**—Clears the DF bit in the new header.
- **set**—Sets the DF bit in the new header.
- **copy**—Copies the DF bit in the original IP header to the new IP header.

You can configure the DF bit in IPsec policy view, IPsec policy template view, IPsec profile view, interface view, and system view. The DF bit setting in IPsec policy view, IPsec policy template view, or IPsec profile view has the highest priority. If the DF bit setting is not configured in the IPsec policy, IPsec profile, or IPsec policy template, the interface-view DF bit setting is used. If the DF bit setting is not configured in interface view, the global DF bit setting configured in system view is used.

Restrictions and guidelines for DF bit configuration for IPsec packets

The DF bit setting takes effect only in tunnel mode, and it changes the DF bit in the new IP header rather than the original IP header.

Only IKE-based IPsec supports configuring the DF bit.

Configure the same DF bit setting on the interfaces where the same IPsec policy bound to a source interface is applied.

If the DF bit is set, the devices on the path cannot fragment the IPsec packets. To prevent IPsec packets from being discarded, make sure the path MTU is larger than the IPsec packet size. As a best practice, clear the DF bit if you cannot make sure the path MTU is larger than the IPsec packet size.

Configuring the DF bit of IPsec packets in an IPsec profile, IPsec policy or IPsec policy template

1. Enter system view.
   system-view
2. Enter IPsec profile, IPsec policy or IPsec policy template view.
   - Enter IPsec profile view.
     ipsec profile *profile-name* **isakmp**

- Enter IPsec policy view.

  ipsec { **ipv6-policy** | **policy** } *policy-name seq-number* **isakmp**
- Enter IPsec policy template view.

  ipsec { **ipv6-policy-template** | **policy-template** }*template-name seq-number*

3. Configure the DF bit of IPsec packets.

   ipsec df-bit { **clear** | **copy** | **set** }

   By default, an IPsec profile, IPsec policy or IPsec policy template uses the interface-specific or global DF bit setting.

Configuring the DF bit of IPsec packets on an interface

1. Enter system view.

   system-view
2. Enter interface view.

   interface *interface-type interface-number*
3. Configure the DF bit of IPsec packets on the interface.

   ipsec df-bit { **clear** | **copy** | **set** }

   By default, the interface uses the global DF bit setting.

Configuring the DF bit of IPsec packets globally

1. Enter system view.

   system-view
2. Configure the DF bit of IPsec packets globally.

   ipsec global-df-bit { **clear** | **copy** | **set** }

   By default, IPsec copies the DF bit in the original IP header to the new IP header.

Setting the maximum number of IPsec tunnels

Restrictions and guidelines

To maximize concurrent performance of IPsec when memory is sufficient, increase the maximum number of IPsec tunnels. To ensure service availability when memory is insufficient, decrease the maximum number of IPsec tunnels.

Procedure

1. Enter system view.

   system-view
2. Set the maximum number of IPsec tunnels.

   ipsec limit max-tunnel *tunnel-limit*

   By default, the number of IPsec tunnels is not limited.

Enabling logging for IPsec packets

About this task

Perform this task to enable logging for IPsec packets that are discarded for reasons such as IPsec SA lookup failure, AH-ESP authentication failure, and ESP encryption failure. The log information includes the source and destination IP addresses, SPI value, and sequence number of a discarded IPsec packet, and the reason for the discard.

Procedure

1. Enter system view.

   system-view

2. Enable logging for IPsec packets.

   ipsec logging packet enable

   By default, logging for IPsec packets is disabled.

Enabling logging for IPsec negotiation

About this task

This feature enables the device to output logs for the IPsec negotiation process.

Procedure

1. Enter system view.

   system-view

2. Enable logging for IPsec negotiation.

   ipsec logging negotiation enable

   By default, logging for IPsec negotiation is enabled.

Configuring SNMP notifications for IPsec

About this task

After you enable SNMP notifications for IPsec, the IPsec module notifies the NMS of important module events. The notifications are sent to the device's SNMP module. For the notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP notifications, see *Network Management and Monitoring Configuration Guide*.

To generate and output SNMP notifications for a specific IPsec failure or event type, perform the following tasks:

1. Enable SNMP notifications for IPsec globally.
2. Enable SNMP notifications for the failure or event type.

Procedure

1. Enter system view.

   system-view

2. Enable SNMP notifications for IPsec globally.

   snmp-agent trap enable ipsec global

   By default, SNMP notifications for IPsec are disabled.

3. Enable SNMP notifications for the specified failure or event types.

   snmp-agent trap enable ipsec [**auth-failure** | **connection-start** | **connection-stop** | **decrypt-failure** | **encrypt-failure** | **invalid-sa-failure** | **no-sa-failure** | **policy-add** | **policy-attach** | **policy-delete** | **policy-detach** | **tunnel-start** | **tunnel-stop**] *

   By default, SNMP notifications for all failure and event types are disabled.

Display and maintenance commands for IPsec

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPsec policy information.	**display ipsec** { **ipv6-policy** | **policy** } [*policy-name* [*seq-number*]]
Display IPsec policy template information.	**display ipsec** { **ipv6-policy-template** | **policy-template** } [*template-name* [*seq-number*]]
Display IPsec profile information.	**display ipsec profile** [*profile-name*]
Display IPsec SA information.	**display ipsec sa** [**brief** | **count** | **interface** *interface-type interface-number* | { **ipv6-policy** | **policy** } *policy-name* [*seq-number*] | **profile** *profile-name* | **remote** [**ipv6**] *ip-address*]
Display IPsec smart link policy information.	**display ipsec smart-link policy** [**brief** | **name** *policy-name*]
Display IPsec statistics.	**display ipsec statistics** [**tunnel-id** *tunnel-id*]
Display IPsec transform set information.	**display ipsec transform-set** [*transform-set-name*]
Display IPsec tunnel information.	**display ipsec tunnel** { **brief** | **count** | **tunnel-id** *tunnel-id* }
Clear IPsec SAs.	**reset ipsec sa** [{ **ipv6-policy** | **policy** } *policy-name* [*seq-number*] | **profile** *profile-name* | **remote** { *ipv4-address* | **ipv6** *ipv6-address* } | **spi** { *ipv4-address* | **ipv6** *ipv6-address* } { **ah** | **esp** } *spi-num*]
Clear IPsec statistics.	**reset ipsec statistics** [**tunnel-id** *tunnel-id*]

IPsec configuration examples

Example: Configuring a manual mode IPsec tunnel for IPv4 packets between gateways

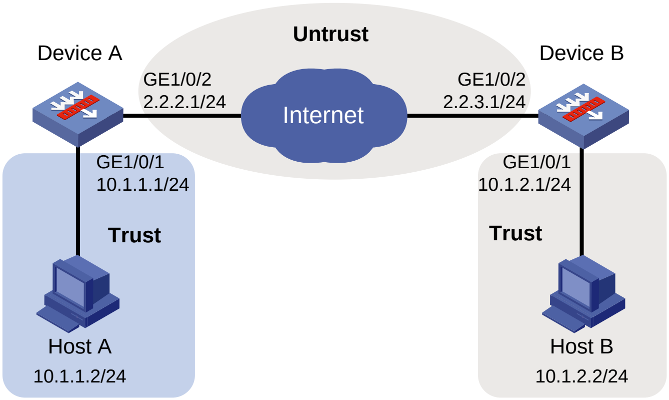
Network configuration

As shown in Figure 10, establish an IPsec tunnel between Device A and Device B to protect data flows between subnet 10.1.1.0/24 and subnet 10.1.2.0/24. Configure the tunnel as follows:

- Specify the encapsulation mode as tunnel, the security protocol as ESP, the encryption algorithm as 128-bit AES, and the authentication algorithm as HMAC-SHA1.

- Manually set up IPsec SAs.

Figure 10 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 2.2.2.2
   [DeviceA] ip route-static 2.2.3.1 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.3.1
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.3.1
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flow to be protected:

# Configure an IPv4 advanced ACL to identify the data flow from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

# Create a manual IPsec policy entry, which specifies the ACL for IPsec, the IPsec transform set, and the remote IP address of the IPsec tunnel.

```
[DeviceA] ipsec policy map1 10 manual
[DeviceA-ipsec-policy-manual-map1-10] security acl 3101
```

```
[DeviceA-ipsec-policy-manual-map1-10] transform-set tran1
[DeviceA-ipsec-policy-manual-map1-10] remote-address 2.2.3.1
[DeviceA-ipsec-policy-manual-map1-10] sa spi outbound esp 12345
[DeviceA-ipsec-policy-manual-map1-10] sa spi inbound esp 54321
[DeviceA-ipsec-policy-manual-map1-10] sa string-key outbound esp simple abcdefg
[DeviceA-ipsec-policy-manual-map1-10] sa string-key inbound esp simple gfedcba
[DeviceA-ipsec-policy-manual-map1-10] quit
```

8. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/2] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.2.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.3.2.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.3.2
[DeviceB] ip route-static 2.2.2.1 24 2.2.3.2
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
```

```
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure an IPv4 advanced ACL to identify data flows from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

# Create a manual IPsec policy entry, which specifies the ACL for IPsec, the IPsec transform set, and the remote IP address of the IPsec tunnel.

```
[DeviceB] ipsec policy use1 10 manual
[DeviceB-ipsec-policy-manual-use1-10] security acl 3101
[DeviceB-ipsec-policy-manual-use1-10] transform-set tran1
[DeviceB-ipsec-policy-manual-use1-10] remote-address 2.2.2.1
[DeviceB-ipsec-policy-manual-use1-10] sa spi outbound esp 54321
[DeviceB-ipsec-policy-manual-use1-10] sa spi inbound esp 12345
```

```
[DeviceB-ipsec-policy-manual-use1-10] sa string-key outbound esp simple gfedcba
[DeviceB-ipsec-policy-manual-use1-10] sa string-key inbound esp simple abcdefg
[DeviceB-ipsec-policy-manual-use1-10] quit
```

8. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipsec policy use1
[DeviceB-GigabitEthernet1/0/2] quit
```

Verifying the configuration

After the configuration is completed, an IPsec tunnel between Device A and Device B is established, and the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24 is IPsec-protected. This example uses Device A to verify the configuration.

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet 1/0/2
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: Manual
  -----------------------------
    Tunnel id: 549
    Encapsulation mode: tunnel
    Transmitting entity: Initiator
    Path MTU: 1443
    Tunnel:
        local  address: 2.2.2.1
        remote address: 2.2.3.1
    Flow:
        as defined in ACL 3101
    [Inbound ESP SA]
      SPI: 54321 (0x0000d431)
      Connection ID: 1
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      No duration limit for this SA
    [Outbound ESP SA]
      SPI: 12345 (0x00003039)
      Connection ID: 2
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      No duration limit for this SA
```

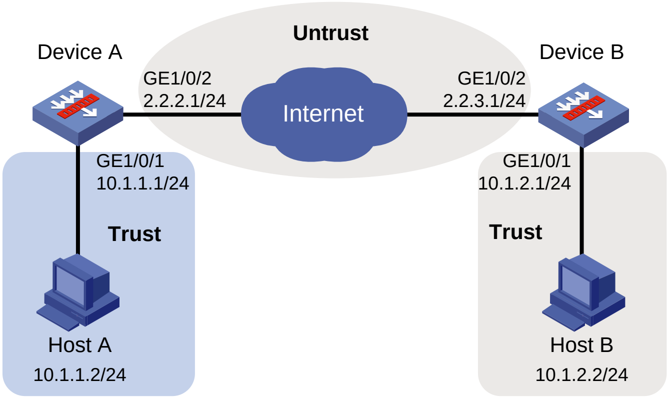
Example: Configuring an IKE-based IPsec tunnel for IPv4 packets between gateways (preshared key authentication)

Network configuration

As shown in Figure 11, establish an IPsec tunnel between Device A and Device B to protect data flows between subnet 10.1.1.0/24 and subnet 10.1.2.0/24. Configure the IPsec tunnel as follows:

- Specify the encapsulation mode as tunnel, the security protocol as ESP, the encryption algorithm as 128-bit AES, and the authentication algorithm as HMAC-SHA1.
- Set up SAs through IKE negotiation.

Figure 11 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 2.2.2.2
   [DeviceA] ip route-static 2.2.3.1 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.3.1
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.3.1
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure an IPv4 advanced ACL to identify data flows from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
```

```
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceA] ike keychain keychain1
   [DeviceA-ike-keychain-keychain1] pre-shared-key address 2.2.3.1 255.255.255.0 key
   simple 123456TESTplat&!
   [DeviceA-ike-keychain-keychain1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceA] ike profile profile1
   [DeviceA-ike-profile-profile1] keychain keychain1
   [DeviceA-ike-profile-profile1] match remote identity address 2.2.3.1 255.255.255.0
   [DeviceA-ike-profile-profile1] quit
   ```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceA] ipsec policy map1 10 isakmp
   [DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
   [DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
   [DeviceA-ipsec-policy-isakmp-map1-10] local-address 2.2.2.1
   [DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.3.1
   [DeviceA-ipsec-policy-isakmp-map1-10] ike-profile profile1
   [DeviceA-ipsec-policy-isakmp-map1-10] quit
   ```

10. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

    ```
    [DeviceA] interface gigabitethernet 1/0/2
    [DeviceA-GigabitEthernet1/0/2] ipsec apply policy map1
    [DeviceA-GigabitEthernet1/0/2] quit
    ```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 10.1.2.1 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.3.2.

   ```
   [DeviceB] ip route-static 10.1.1.0 24 2.2.3.2
   [DeviceB] ip route-static 2.2.2.1 24 2.2.3.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure an IPv4 advanced ACL to identify data flows from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceB] ike keychain keychain1
   [DeviceB-ike-keychain-keychain1] pre-shared-key address 2.2.2.1 255.255.255.0 key
   simple 123456TESTplat&!
   [DeviceB-ike-keychain-keychain1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceB] ike profile profile1
   [DeviceB-ike-profile-profile1] keychain keychain1
   [DeviceB-ike-profile-profile1] match remote identity address 2.2.2.1 255.255.255.0
   [DeviceB-ike-profile-profile1] quit
   ```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceB] ipsec policy use1 10 isakmp
   [DeviceB-ipsec-policy-isakmp-use1-10] security acl 3101
   [DeviceB-ipsec-policy-isakmp-use1-10] transform-set tran1
   [DeviceB-ipsec-policy-isakmp-use1-10] local-address 2.2.3.1
   [DeviceB-ipsec-policy-isakmp-use1-10] remote-address 2.2.2.1
   [DeviceB-ipsec-policy-isakmp-use1-10] ike-profile profile1
   [DeviceB-ipsec-policy-isakmp-use1-10] quit
   ```

10. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

   ```
   [DeviceB] interface gigabitethernet 1/0/2
   [DeviceB-GigabitEthernet1/0/2] ipsec apply policy use1
   [DeviceB-GigabitEthernet1/0/2] quit
   ```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, the traffic between the two subnets is IPsec-protected.

# Display IPsec SAs on Device A and Device B. This example uses Device A to verify the configuration.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet 1/0/2
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
```

```
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1443
    Tunnel:
        local  address: 2.2.3.1
        remote address: 2.2.2.1
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3769702703 (0xe0b1192f)
      Connection ID: 90194313219
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 3000/28800
      SA remaining duration (kilobytes/sec): 2300/797
      Max received sequence-number: 1
      Anti-replay check enable: N
      Anti-replay window size:
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 3840956402 (0xe4f057f2)
      Connection ID: 64424509441
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 3000/28800
      SA remaining duration (kilobytes/sec): 2312/797
      Max sent sequence-number: 1
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

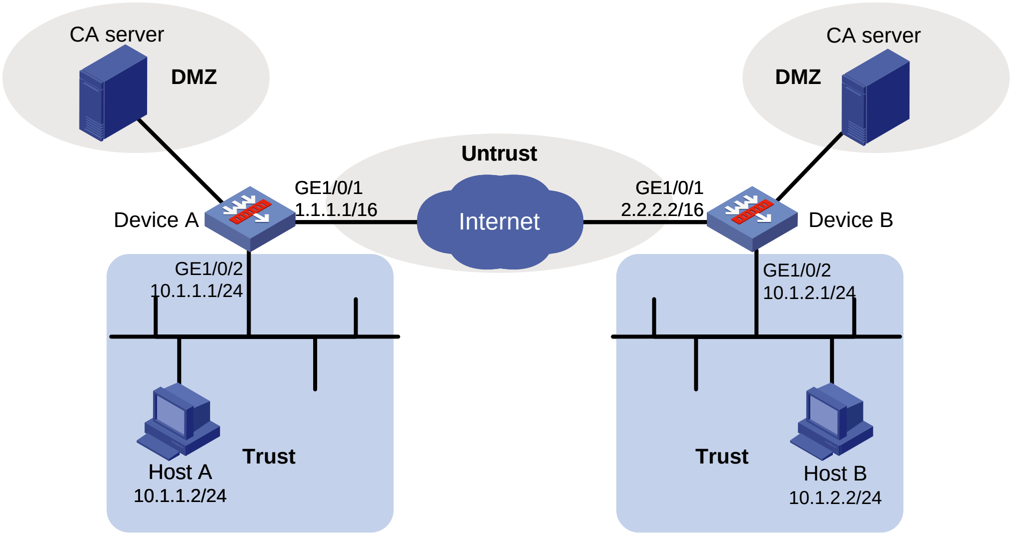
Example: Configuring an IKE-based IPsec tunnel for IPv4 packets between gateways (RSA signature authentication)

Network configuration

As shown in Figure 12, configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use aggressive mode for IKE negotiation phase 1 and to use RSA signature authentication. Device A acts as the initiator, and the subnet where Device A resides uses IP addresses dynamically allocated.

Figure 12 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule name ipseclocalout
   [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
   ```

```
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure a PKI entity.

```
[DeviceA] pki entity entity1
[DeviceA-pki-entity-entity1] common-name devicea
[DeviceA-pki-entity-entity1] quit
```

8. Configure a PKI domain for certificate requests.

```
[DeviceA] pki domain domain1
[DeviceA-pki-domain-domain1] public-key rsa general name rsa1
[DeviceA-pki-domain-domain1] undo crl check enable
[DeviceA-pki-domain-domain1] quit
[DeviceA] pki import domain domain1 der ca filename ca.cer
[DeviceA] pki import domain domain1 p12 local filename server.pfx
```

9. Configure a certificate-based access control policy to control user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceA] pki certificate access-control-policy policy1
[DeviceA-pki-cert-acp-policy1] rule 1 permit group1
[DeviceA] pki certificate attribute-group group1
[DeviceA-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

10. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] certificate domain domain1
[DeviceA-ike-profile-profile1] exchange-mode aggressive
[DeviceA-ike-profile-profile1] local-identity dn
[DeviceA-ike-profile-profile1] match remote certificate policy1
[DeviceA-ike-profile-profile1] quit
```

11. Configure an IKE proposal to specify the parameters used for IKE negotiation.

```
[DeviceA] ike proposal 10
[DeviceA-ike-proposal-10] authentication-algorithm md5
[DeviceA-ike-proposal-10] authentication-method rsa-signature
[DeviceA-ike-proposal-10] quit
```

12. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data:

```
[DeviceA] ipsec policy map1 10 isakmp
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
[DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
[DeviceA-ipsec-policy-isakmp-map1-10] ike-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

13. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
```

```
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
```

```
   [DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
   [DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
   [DeviceB-security-policy-ip-4-untrust-trust] action pass
   [DeviceB-security-policy-ip-4-untrust-trust] quit
   [DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

   ```
   [DeviceB] acl advanced 3101
   [DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
   10.1.1.0 0.0.0.255
   [DeviceB-acl-ipv4-adv-3101] quit
   ```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

   ```
   [DeviceB] ipsec transform-set tran1
   [DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
   [DeviceB-ipsec-transform-set-tran1] protocol esp
   [DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
   [DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
   [DeviceB-ipsec-transform-set-tran1] quit
   ```

7. Configure a PKI entity.

   ```
   [DeviceB] pki entity entity2
   [DeviceB-pki-entity-entity2] common-name deviceb
   [DeviceB-pki-entity-entity2] quit
   ```

8. Configure a PKI domain for certificate request.

   ```
   [DeviceB] pki domain domain2
   [DeviceB-pki-domain-domain2] public-key rsa general name rsa1
   [DeviceB-pki-domain-domain2] undo crl check enable
   [DeviceB-pki-domain-domain2] quit
   [DeviceB] pki import domain domain2 der ca filename ca.cer
   [DeviceB] pki import domain domain2 p12 local filename server.pfx
   ```

9. Configure a certificate-based access control policy to control user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

   ```
   [DeviceB] pki certificate access-control-policy policy1
   [DeviceB-pki-cert-acp-policy1] rule 1 permit group1
   [DeviceB] pki certificate attribute-group group1
   [DeviceB-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
   ```

10. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

    ```
    [DeviceB] ike profile profile2
    [DeviceB-ike-profile-profile2] certificate domain domain2
    [DeviceB-ike-profile-profile2] exchange-mode aggressive
    [DeviceB-ike-profile-profile2] local-identity dn
    [DeviceB-ike-profile-profile2] match remote certificate policy1
    [DeviceB-ike-profile-profile2] quit
    ```

11. Configure an IKE proposal to specify the security parameters for IKE negotiation.

    ```
    [DeviceB] ike proposal 10
    ```

```
[DeviceB-ike-proposal-10] authentication-algorithm md5
[DeviceB-ike-proposal-10] authentication-method rsa-signature
[DeviceB-ike-proposal-10] quit
```

12. Configure an IPsec policy template, which is used to create an IPsec policy:

    # Create an IPsec policy template entry. Specify **template1** as the template name and set the sequence number to 1.

    ```
    [DeviceB] ipsec policy-template template1 1
    [DeviceB-ipsec-policy-template-template1-1] transform-set tran1
    [DeviceB-ipsec-policy-template-template1-1] ike-profile profile2
    [DeviceB-ipsec-policy-template-template1-1] quit
    ```

13. Create an IKE-based IPsec policy entry by using IPsec policy template **template1**, so as to establish the IPsec tunnel to protect data.

    ```
    [DeviceB] ipsec policy use1 1 isakmp template template1
    ```

14. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/1
    [DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
    [DeviceB-GigabitEthernet1/0/1] quit
    ```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, traffic between the two subnets is IPsec-protected.

# Display the IKE SA on Device A.

```
[DeviceA] display ike sa
    Connection-ID   Remote                Flag         DOI
------------------------------------------------------------------
    13              2.2.2.2/500           RD           IPsec
Flags:
RD--READY RL--REPLACED FD-FADING RK-REKEY
[DeviceA] display ike sa verbose
   -----------------------------------------------
   Connection ID: 13
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Initiator
   Initiator cookie: 1bcf453f0a217259
   Responder cookie: 5e32a74dfa66a0a4
   -----------------------------------------------
   Local IP/port: 1.1.1.1/500
   Local ID type: FQDN
   Local ID: www.devicea.com

   Remote IP/port: 2.2.2.2/500
   Remote ID type: IPV4_ADDR
   Remote ID: 2.2.2.2

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: SHA1
   Encryption-algorithm: DES-CBC
```

```
   Life duration(sec): 86400
   Remaining key duration(sec): 84565
   Exchange-mode: Aggressive
   Diffie-Hellman group: Group 1
   NAT traversal: Detected

   Extend authentication: Disabled
   Assigned IP address:
   Vendor ID index: 0xa1d
   Vendor ID sequence number: 0x0
```

# Display the IPsec SAs generated on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1435
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 830667426 (0x3182faa2)
      Connection ID: 90194313219
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: Y
      Status: Active
```

```
[Outbound ESP SAs]
  SPI: 3516214669 (0xd1952d8d)
  Connection ID: 64424509441
  Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
  SA duration (kilobytes/sec): 1843200/3600
  SA remaining duration (kilobytes/sec): 1843200/2313
  Max sent sequence-number:
  UDP encapsulation used for NAT traversal: Y
  Status: Active
```

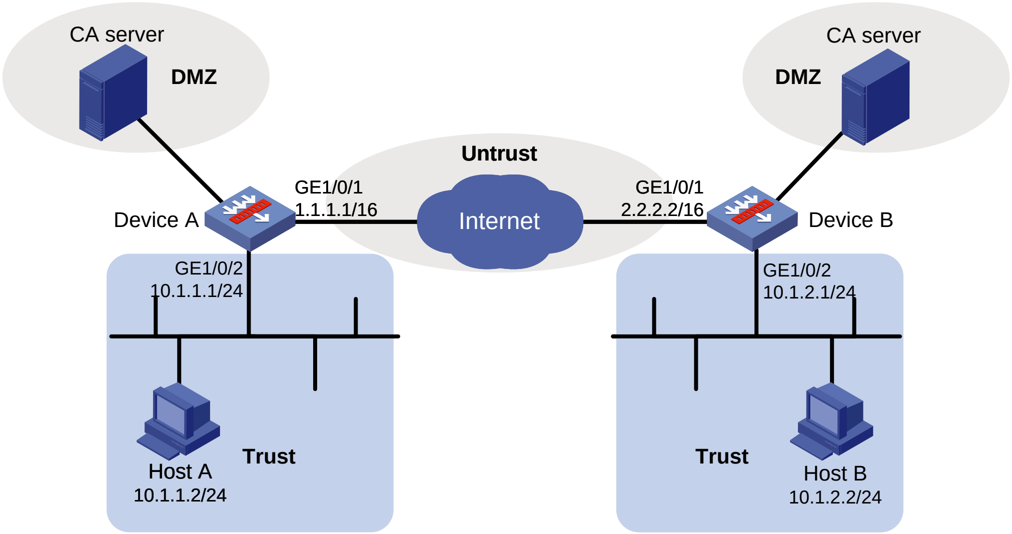
Example: Configuring an IKE-based IPsec tunnel for IPv4 packets between gateways (SM2-DE digital envelop authentication)

Network configuration

As shown in Figure 13, configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use GM main mode and SM2-DE digital envelop authentication for the IKE negotiation phase 1.

Figure 13 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. **Configure settings for routing.**

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

```
[DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
[DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host A and Host B:

   # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
```

```
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. **5.** Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

   ```
   [DeviceA] acl advanced 3101
   [DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
   10.1.2.0 0.0.0.255
   [DeviceA-acl-ipv4-adv-3101] quit
   ```

6. **6.** Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

   ```
   [DeviceA] ipsec transform-set tran1
   [DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
   [DeviceA-ipsec-transform-set-tran1] protocol esp
   [DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm sm1-cbc-128
   [DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sm3
   [DeviceA-ipsec-transform-set-tran1] quit
   ```

7. **7.** Configure a PKI entity.

   ```
   [DeviceA] pki entity entity1
   [DeviceA-pki-entity-entity1] common-name devicea
   [DeviceA-pki-entity-entity1] quit
   ```

8. **8.** Configure a PKI domain for certificate request.

   ```
   [DeviceA] pki domain domain1
   [DeviceA-pki-domain-domain1] public-key sm2 general name sm2-1
   [DeviceA-pki-domain-domain1] undo crl check enable
   [DeviceA-pki-domain-domain1] quit
   [DeviceA] pki import domain domain1 der ca filename ca.cer
   [DeviceA] pki import domain domain1 p12 local filename server.pfx
   ```

9. **9.** Configure an IKE proposal to specify the parameters for IKE neogitiation.

   ```
   [DeviceA] ike proposal 10
   [DeviceA-ike-proposal-10] authentication-method sm2-de
   [DeviceA-ike-proposal-10] authentication-algorithm sm3
   [DeviceA-ike-proposal-10] encryption-algorithm sm1-cbc-128
   [DeviceA-ike-proposal-10] quit
   ```

10. **10.** Configure an IKE profile to specify the security parameters used to establish IKE SAs.

    ```
    [DeviceA] ike profile profile1
    [DeviceA-ike-profile-profile1] exchange-mode gm-main
    [DeviceA-ike-profile-profile1] certificate domain domain1
    [DeviceA-ike-profile-profile1] proposal 10
    [DeviceA-ike-profile-profile1] local-identity address 1.1.1.1
    [DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
    [DeviceA-ike-profile-profile1] quit
    ```

11. **11.** Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

    ```
    [DeviceA] ipsec policy map1 10 isakmp
    [DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
    ```

```
[DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
[DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-map1-10] ike-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

12. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
```

```
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm sm1-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sm3
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure a PKI entity.

```
[DeviceB] pki entity entity2
[DeviceB-pki-entity-entity2] common-name deviceb
[DeviceB-pki-entity-entity2] quit
```

8. Configure a PKI domain for certificate request.

```
[DeviceB] pki domain domain2
[DeviceB-pki-domain-domain2] public-key sm2 general name sm2-1
[DeviceB-pki-domain-domain2] undo crl check enable
[DeviceB-pki-domain-domain2] quit
[DeviceB] pki import domain domain2 der ca filename ca.cer
[DeviceB] pki import domain domain2 p12 local filename server.pfx
```

9. Configure an IKE proposal to specify the parameters for IKE neogitiation.

```
[DeviceB] ike proposal 10
[DeviceB-ike-proposal-10] authentication-method sm2-de
[DeviceB-ike-proposal-10] authentication-algorithm sm3
[DeviceB-ike-proposal-10] encryption-algorithm sm1-cbc-128
[DeviceB-ike-proposal-10] quit
```

10. Configure an IKE profile to specify the security parameters used to establish IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] exchange-mode gm-main
[DeviceB-ike-profile-profile1] certificate domain domain2
[DeviceB-ike-profile-profile1] proposal 10
[DeviceB-ike-profile-profile1] local-identity address 2.2.2.2
[DeviceB-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.0.0
[DeviceB-ike-profile-profile1] quit
```

11. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceB] ipsec policy use1 10 isakmp
[DeviceB-ipsec-policy-isakmp-use1-10] remote-address 1.1.1.1
[DeviceB-ipsec-policy-isakmp-use1-10] security acl 3101
[DeviceB-ipsec-policy-isakmp-use1-10] transform-set tran1
[DeviceB-ipsec-policy-isakmp-use1-10] ike-profile profile1
[DeviceB-ipsec-policy-isakmp-use1-10] quit
```

12. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, traffic between the two subnets is IPsec-protected.

# Display the IKE proposal configuration on Device A and Device B.

```
[DeviceA] display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
              method       algorithm    algorithm       group      (seconds)
----------------------------------------------------------------------------
10       SM2-DE             SM3         SM1-CBC-128     Group 1       86400
default  PRE-SHARED-KEY     SHA1        DES-CBC         Group 1       86400

[DeviceB] display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
              method       algorithm    algorithm       group      (seconds)
----------------------------------------------------------------------------
10       SM2-DE             SM3         SM1-CBC-128    Group 1       86400
default  PRE-SHARED-KEY     SHA1        DES-CBC        Group 1       86400
```

# Display the IKE SA on Device A.

```
[DeviceA] display ike sa
    Connection-ID   Remote                Flag         DOI
------------------------------------------------------------------
    1               2.2.2.2/500           RD           IPsec
Flags:
```

```
RD--READY RL--REPLACED FD-FADING RK-REKEY
```

# Display IPsec SAs generated on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1456
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 1451246811 (0x568044db)
      Connection ID: 90194313219
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 2692887942 (0xa0823586)
      Connection ID: 64424509441
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max sent sequence-number:
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Display the IKE SA and IPsec SAs on Device B. (Details not shown.)

```
[DeviceB] display ike sa
[DeviceB] display ipsec sa
```

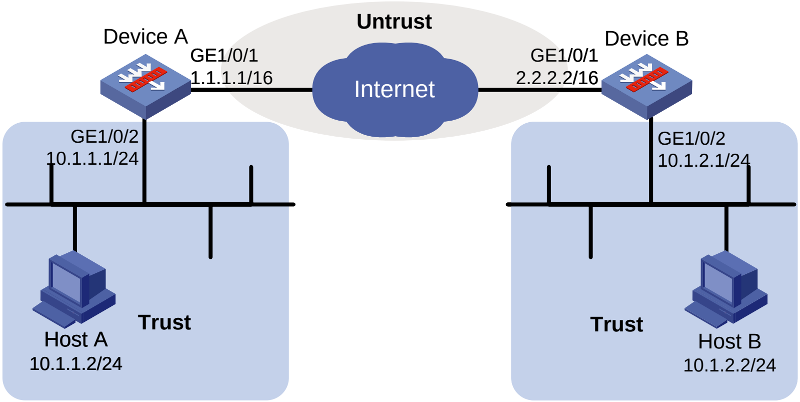
Example: Configuring an IKEv2-based IPsec tunnel for IPv4 packets between gateways (preshared key authentication)

Network configuration

As shown in Figure 14, configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKEv2 proposal and the default IKEv2 policy in IKEv2 negotiation to set up IPsec SAs.
- Configure the two devices to use the preshared key authentication method in IKEv2 negotiation.

Figure 14 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ikev2 keychain keychain1
[DeviceA-ikev2-keychain-keychain1] peer peer1
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 16
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
[DeviceA-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext abcde
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceA-ikev2-keychain-keychain1] quit
```

8. Configure an IKEv2 profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] authentication-method local pre-share
[DeviceA-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceA-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
[DeviceA-ikev2-profile-profile1] quit
```

9. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ipsec policy map1 10 isakmp
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
[DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
[DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-map1-10] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

      # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceB] ikev2 keychain keychain1
[DeviceB-ikev2-keychain-keychain1] peer peer1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] address 1.1.1.1 16
[DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 1.1.1.1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext abcde
[DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceB-ikev2-keychain-keychain1] quit
```

8. Configure an IKEv2 profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ikev2 profile profile1
[DeviceB-ikev2-profile-profile1] authentication-method local pre-share
[DeviceB-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceB-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] match remote identity address 1.1.1.1 255.255.0.0
[DeviceA-ikev2-profile-profile1] quit
```

9. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceB] ipsec policy use1 10 isakmp
[DeviceB-ipsec-policy-isakmp-use1-10] remote-address 1.1.1.1
[DeviceB-ipsec-policy-isakmp-use1-10] security acl 3101
[DeviceB-ipsec-policy-isakmp-use1-10] transform-set tran1
[DeviceB-ipsec-policy-isakmp-use1-10] ikev2-profile profile1
[DeviceB-ipsec-policy-isakmp-use1-10] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec-protected.

# Display the IKEv2 proposal and IKEv2 policy on Device A.

```
[DeviceA] display ikev2 proposal
```

```
IKEv2 proposal : default
  Encryption: AES-CBC-128 3DES-CBC
  Integrity: SHA1 MD5
  PRF: SHA1 MD5
  DH Group: MODP1536/Group5 MODP1024/Group2
[DeviceA] display ikev2 policy
IKEv2 policy : default
  Match VRF : any
  Proposal: default
```

# Display the IKEv2 SA on Device A.

```
[DeviceA] display ikev2 sa
Tunnel ID   Local                       Remote                      Status
---------------------------------------------------------------------------
  1        1.1.1.1/500                 2.2.2.2/500                   EST
Status:
IN-NEGO: Negotiating, EST: Established, DEL:Deleting
```

# Display the IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Path MTU: 1456
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3264152513 (0xc28f03c1)
      Connection ID: 141733920771
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
```

```
    Anti-replay check enable: Y
    Anti-replay window size: 64
    UDP encapsulation used for NAT traversal: N
    Status: Active

  [Outbound ESP SAs]
    SPI: 738451674 (0x2c03e0da)
    Connection ID: 64424509441
    Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
    SA duration (kilobytes/sec): 1843200/3600
    SA remaining duration (kilobytes/sec): 1843200/3484
    Max sent sequence-number:
    UDP encapsulation used for NAT traversal: N
    Status: Active
```

# Display the IKEv2 proposal, IKEv2 policy, IKEv2 SA and IPsec SAs on Device B.

```
[DeviceB] display ikev2 proposal
[DeviceB] display ikev2 policy
[DeviceB] display ikev2 sa
[DeviceB] display ipsec sa
```

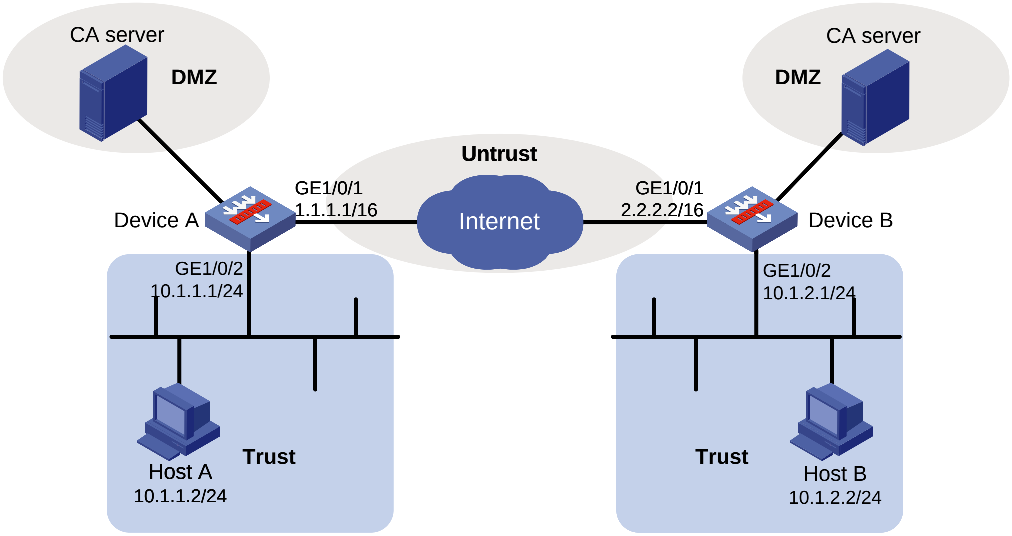
Example: Configuring an IKEv2-based IPsec tunnel for IPv4 packets between gateways (RSA signature authentication)

Network configuration

As shown in Figure 15, configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use IKEv2 negotiation and RSA signature authentication. Device A acts as the initiator, and the subnet where Device A resides uses IP addresses dynamically allocated.

Figure 15 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      ```

```
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure a PKI entity.

```
[DeviceA] pki entity entity1
[DeviceA-pki-entity-entity1] common-name devicea
[DeviceA-pki-entity-entity1] quit
```

8. Configure a PKI domain for certificate request.

```
[DeviceA] pki domain domain1
[DeviceA-pki-domain-domain1] public-key rsa general name rsa1
[DeviceA-pki-domain-domain1] undo crl check enable
[DeviceA-pki-domain-domain1] quit
[DeviceA] pki import domain domain1 der ca filename ca.cer
[DeviceA] pki import domain domain1 p12 local filename server.pfx
```

9. Configure a certificate-based access control policy to control the user access rights.

A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceA] pki certificate access-control-policy policy1
[DeviceA-pki-cert-acp-policy1] rule 1 permit group1
[DeviceA] pki certificate attribute-group group1
```

```
[DeviceA-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

10. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] authentication-method local rsa-signature
[DeviceA-ikev2-profile-profile1] authentication-method remote rsa-signature
[DeviceA-ikev2-profile-profile1] certificate domain domain1
[DeviceA-ikev2-profile-profile1] local-identity dn
[DeviceA-ikev2-profile-profile1] match remote certificate policy1
[DeviceA-ikev2-profile-profile1] quit
```

11. Configure an IKEv2 proposal to specify the parameters for IKE negotiation.

```
[DeviceA] ikev2 proposal 10
[DeviceA-ikev2-proposal-10] integrity md5
[DeviceA-ikev2-proposal-10] encryption 3des-cbc
[DeviceA-ikev2-proposal-10] dh group1
[DeviceA-ikev2-proposal-10] prf md5
[DeviceA-ikev2-proposal-10] quit
```

12. Configure an IKEv2 policy, which is used to negotiate IKEv2 SAs.

```
[DeviceA] ikev2 policy 1
[DeviceA-ikev2-policy-1] proposal 10
[DeviceA-ikev2-policy-1] quit
```

13. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ipsec policy map1 10 isakmp
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
[DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
[DeviceA-ipsec-policy-isakmp-map1-10] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

14. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] action pass
      [DeviceB-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] action pass
      [DeviceB-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

      ```
      [DeviceB-security-policy-ip] rule name untrust-trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-4-untrust-trust] action pass
      [DeviceB-security-policy-ip-4-untrust-trust] quit
      [DeviceB-security-policy-ip] quit
      ```

5. Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

   ```
   [DeviceB] acl advanced 3101
   ```

```
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure a PKI entity.

```
[DeviceB] pki entity entity2
[DeviceB-pki-entity-entity2] common-name deviceb
[DeviceB-pki-entity-entity2] quit
```

8. Configure a PKI domain for certificate request.

```
[DeviceB] pki domain domain2
[DeviceB-pki-domain-domain2] public-key rsa general name rsa1
[DeviceB-pki-domain-domain2] undo crl check enable
[DeviceB-pki-domain-domain2] quit
[DeviceB] pki import domain domain2 der ca filename ca.cer
[DeviceB] pki import domain domain2 p12 local filename server.pfx
```

9. Configure a certificate-based access control policy to control the user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceB] pki certificate access-control-policy policy1
[DeviceB-pki-cert-acp-policy1] rule 1 permit group1
[DeviceB] pki certificate attribute-group group1
[DeviceB-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

10. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceB] ikev2 profile profile2
[DeviceB-ikev2-profile-profile2] authentication-method local rsa-signature
[DeviceB-ikev2-profile-profile2] authentication-method remote rsa-signature
[DeviceB-ikev2-profile-profile2] certificate domain domain2
[DeviceB-ikev2-profile-profile2] local-identity dn
[DeviceB-ikev2-profile-profile2] match remote certificate policy1
[DeviceB-ikev2-profile-profile2] quit
```

11. Configure an IKEv2 proposal to specify the parameters for IKE negotiation.

```
[DeviceB] ikev2 proposal 10
[DeviceB-ikev2-proposal-10] integrity md5
[DeviceB-ikev2-proposal-10] encryption 3des-cbc
[DeviceB-ikev2-proposal-10] dh group1
[DeviceB-ikev2-proposal-10] prf md5
[DeviceB-ikev2-proposal-10] quit
```

12. Configure an IKEv2 policy, which is used to negotiate IKEv2 SAs.

```
[DeviceB] ikev2 policy 1
[DeviceB-ikev2-policy-1] proposal 10
```

```
[DeviceB-ikev2-policy-1] quit
```

13. Configure an IPsec policy template, which is used to create the IPsec policy.

```
[DeviceB] ipsec policy-template template1 1
[DeviceB-ipsec-policy-template-template1-1] remote-address 1.1.1.1
[DeviceB-ipsec-policy-template-template1-1] security acl 3101
[DeviceB-ipsec-policy-template-template1-1] transform-set tran1
[DeviceB-ipsec-policy-template-template1-1] ikev2-profile profile2
[DeviceB-ipsec-policy-template-template1-1] quit
```

14. Create an IKE-based IPsec policy entry by using IPsec policy template **template1**, so as to establish the IPsec tunnel to protect the specified data flow.

```
[DeviceB] ipsec policy use1 1 isakmp template template1
```

15. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec-protected.

# Display the IKEv2 proposal configuration on Device A and Device B.

```
[DeviceA] display ikev2 proposal 10
IKEv2 proposal : 10
  Encryption : 3DES-CBC
  Integrity : MD5
  PRF : MD5
  DH Group : MODP768/Group1
[DeviceB] display ikev2 proposal 10
IKEv2 proposal : 10
  Encryption : 3DES-CBC
  Integrity : MD5
  PRF : MD5
  DH Group : MODP768/Group1
```

# Display the IKEv2 policy configuration Device A and Device B.

```
[DeviceA] display ikev2 policy 1
IKEv2 policy : 1
  Priority: 100
  Match Local : any
  Match VRF : public
  Proposal : 10
[DeviceB] display ikev2 policy 1
IKEv2 policy : 1
  Priority: 100
  Match Local : any
  Match VRF : public
  Proposal : 10
```

# Display the IKEv2 SA on Device A.

```
[DeviceA] display ikev2 sa
```

```
Tunnel ID   Local                       Remote                      Status
---------------------------------------------------------------------------
  1        1.1.1.1/500                 2.2.2.2/500                  EST
Status:
IN-NEGO: Negotiating, EST: Established, DEL:Deleting
```

# Display information about the CA certificate on Device A.

```
[DeviceA] display pki certificate domain domain1 ca
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            b9:14:fb:25:c9:08:2c:9d:f6:94:20:30:37:4e:00:00
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=cn, O=rnd, OU=sec, CN=8088
        Validity
            Not Before: Sep  6 01:53:58 2012 GMT
            Not After : Sep  8 01:50:58 2015 GMT
        Subject: C=cn, O=rnd, OU=sec, CN=8088
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:de:81:f4:42:c6:9f:c2:37:7b:21:84:57:d6:42:
                    00:69:1c:4c:34:a4:5e:bb:30:97:45:2b:5e:52:43:
                    c0:49:1f:e1:d8:0f:5c:48:c2:39:69:d1:84:e4:14:
                    70:3d:98:41:28:1c:20:a1:9a:3f:91:67:78:77:27:
                    d9:08:5f:7a:c4:36:45:8b:f9:7b:e7:7d:6a:98:bb:
                    4e:a1:cb:2c:3d:92:66:bd:fb:80:35:16:c6:35:f0:
                    ff:0b:b9:3c:f3:09:94:b7:d3:6f:50:8d:83:f1:66:
                    2f:91:0b:77:a5:98:22:b4:77:ac:84:1d:03:8e:33:
                    1b:31:03:78:4f:77:a0:db:af
                Exponent: 65537 (0x10001)
    Signature Algorithm: sha1WithRSAEncryption
        9a:6d:8c:46:d3:18:8a:00:ce:12:ee:2b:b0:aa:39:5d:3f:90:
        08:49:b9:a9:8f:0d:6e:7b:e1:00:fb:41:f5:d4:0c:e4:56:d8:
        7a:a7:61:1d:2b:b6:72:e3:09:0b:13:9d:fa:c8:fc:c4:65:a7:
        f9:45:21:05:75:2c:bf:36:7b:48:b4:4a:b9:fe:87:b9:d8:cf:
        55:16:87:ec:07:1d:55:5a:89:74:73:68:5e:f9:1d:30:55:d9:
        8a:8f:c5:d4:20:7e:41:a9:37:57:ed:8e:83:a7:80:2f:b8:31:
        57:3a:f2:1a:28:32:ea:ea:c5:9a:55:61:6a:bc:e5:6b:59:0d:
        82:16
```

# Display the local certificate on Device A.

```
[DeviceA]display pki certificate domain domain1 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            a1:f4:d4:fd:cc:54:c3:07:c4:9e:15:2d:5f:64:57:77
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=cn, O=rnd, OU=sec, CN=8088
        Validity
            Not Before: Sep 26 02:06:43 2012 GMT
            Not After : Sep 26 02:06:43 2013 GMT
        Subject: CN=devicea
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:b0:a1:cd:24:6e:1a:1d:51:79:f0:2a:3e:9f:e9:
                    84:07:16:78:49:1b:7d:0b:22:f0:0a:ed:75:91:a4:
                    17:fd:c7:ef:d0:66:5c:aa:e3:2a:d9:71:12:e4:c6:
                    25:77:f0:1d:97:bb:92:a8:bd:66:f8:f8:e8:d5:0d:
                    d2:c8:01:dd:ea:e6:e0:80:ad:db:9d:c8:d9:5f:03:
                    2d:22:07:e3:ed:cc:88:1e:3f:0c:5e:b3:d8:0e:2d:
                    ea:d6:c6:47:23:6a:11:ef:3c:0f:6b:61:f0:ca:a1:
                    79:a0:b1:02:1a:ae:8c:c9:44:e0:cf:d1:30:de:4c:
                    f0:e5:62:e7:d0:81:5d:de:d3
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 CRL Distribution Points:

                Full Name:
                  URI:http://xx.rsa.com:447/8088.crl

    Signature Algorithm: sha1WithRSAEncryption
        73:ac:66:f9:b8:b5:39:e1:6a:17:e4:d0:72:3e:26:9e:12:61:
        9e:c9:7a:86:6f:27:b0:b9:a3:5d:02:d9:5a:cb:79:0a:12:2e:
        cb:e7:24:57:e6:d9:77:12:6b:7a:cf:ee:d6:17:c5:5f:d2:98:
        30:e0:ef:00:39:4a:da:ff:1c:29:bb:2a:5b:60:e9:33:8f:78:
        f9:15:dc:a5:a3:09:66:32:ce:36:cd:f0:fe:2f:67:e5:72:e5:
        21:62:85:c4:07:92:c8:f1:d3:13:9c:2e:42:c1:5f:0e:8f:ff:
        65:fb:de:7c:ed:53:ab:14:7a:cf:69:f2:42:a4:44:7c:6e:90:
        7e:cd
```

# Display the IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Path MTU: 1456
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3264152513 (0xc28f03c1)
      Connection ID: 141733920771
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 738451674 (0x2c03e0da)
      Connection ID: 141733920770
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max sent sequence-number:
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Display the information about the CA certificate, local certificate, IKEv2 SA, and IPsec SA on Device B.

```
[DeviceB] display ikev2 sa
[DeviceB] display pki certificate domain domain2 ca
[DeviceB] display pki certificate domain domain2 local
[DeviceB] display ipsec sa
```

Example: Configuring an IKE-based IPsec NAT traversal tunnel for IPv4 packets between gateways

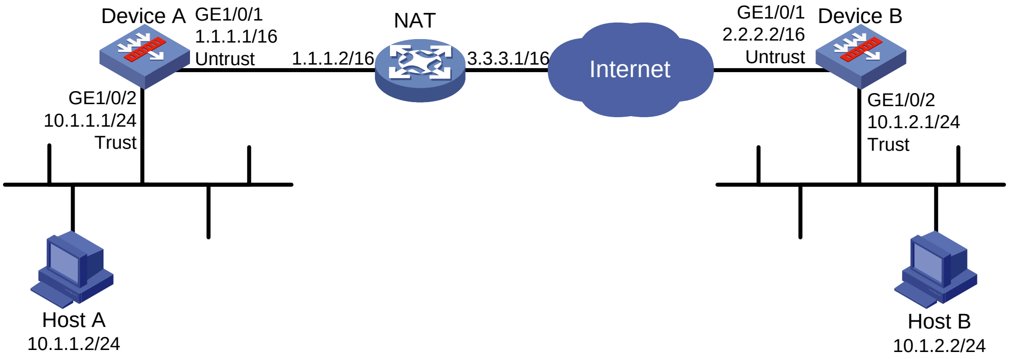
Network configuration

Device A is behind the NAT device. Hosts behind Device A use public IP address 3.3.3.1 to access the external network.

Configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKE proposal for the aggressive IKE negotiation to set up the IPsec SAs.
- Configure the two devices to use the preshared key authentication method for the IKE negotiation phase 1.

Figure 16 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      ```

```
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3000 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3000
[DeviceA-acl-ipv4-adv-3000] rule 0 permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3000] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set transform1
[DeviceA-ipsec-transform-set-transform1] protocol esp
[DeviceA-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceA-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceA-ipsec-transform-set-transform1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ike keychain keychain1
[DeviceA-ike-keychain-keychain1] pre-shared-key address 2.2.2.2 255.255.0.0 key
simple 12345zxcvb!@#$%ZXCVB
```

```
[DeviceA-ike-keychain-keychain1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] keychain keychain1
[DeviceA-ike-profile-profile1] exchange-mode aggressive
[DeviceA-ike-profile-profile1] local-identity fqdn www.devicea.com
[DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
[DeviceA-ike-profile-profile1] quit
```

9. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ipsec policy policy1 1 isakmp
[DeviceA-ipsec-policy-isakmp-policy1-1] remote-address 2.2.2.2
[DeviceA-ipsec-policy-isakmp-policy1-1] transform-set transform1
[DeviceA-ipsec-policy-isakmp-policy1-1] security acl 3000
[DeviceA-ipsec-policy-isakmp-policy1-1] ike-profile profile1
[DeviceA-ipsec-policy-isakmp-policy1-1] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
```

```
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

```
[DeviceB] ipsec transform-set transform1
[DeviceB-ipsec-transform-set-transform1] protocol esp
[DeviceB-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceB-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceB-ipsec-transform-set-transform1] quit
```

6. Configure an IKE keychain to specify the key information used for IKE communication.

This example specifies **12345zxcvb!@#$%ZXCVB** in plain text as the preshared key to be used with the remote peer at 1.1.1.1. The source address of packets from 1.1.1.1 is translated into 3.3.3.1 by the NAT device, so IP address of the remote peer is specified as 3.3.3.1.

```
[DeviceB]ike keychain keychain1
[DeviceB-ike-keychain-keychain1] pre-shared-key address 3.3.3.1 255.255.0.0 key
simple 12345zxcvb!@#$%ZXCVB
[DeviceB-ike-keychain-keychain1] quit
```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile1
```

```
[DeviceB-ike-profile-profile1] keychain keychain1
[DeviceB-ike-profile-profile1] exchange-mode aggressive
[DeviceB-ike-profile-profile1] match remote identity fqdn www.devicea.com
[DeviceB-ike-profile-profile1] quit
```

8. Configure an IPsec policy template, which is used to create an IPsec policy.

```
[DeviceB] ipsec policy-template template1 1
[DeviceB-ipsec-policy-template-template1-1] transform-set transform1
[DeviceB-ipsec-policy-template-template1-1] local-address 2.2.2.2
[DeviceB-ipsec-policy-template-template1-1] ike-profile profile1
[DeviceB-ipsec-policy-template-template1-1] quit
```

9. Create an IKE-based IPsec policy entry by using IPsec policy template **template1**, so as to establish an IPsec tunnel to protect the specified data.

```
[DeviceB] ipsec policy policy1 1 isakmp template template1
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, traffic between the two subnets is IPsec-protected.

# Display the IKE SA on Device A.

```
[DeviceA] display ike sa
    Connection-ID   Remote               Flag         DOI
------------------------------------------------------------------
    13              2.2.2.2/500          RD           IPsec
Flags:
RD--READY RL--REPLACED FD-FADING RK-REKEY
[DeviceA] display ike sa verbose
   -----------------------------------------------
   Connection ID: 13
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Initiator
   Initiator cookie: 1bcf453f0a217259
   Responder cookie: 5e32a74dfa66a0a4
   -----------------------------------------------
   Local IP/port: 1.1.1.1/500
   Local ID type: FQDN
   Local ID: www.devicea.com

   Remote IP/port: 2.2.2.2/500
   Remote ID type: IPV4_ADDR
   Remote ID: 2.2.2.2

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: SHA1
   Encryption-algorithm: DES-CBC
   Life duration(sec): 86400
   Remaining key duration(sec): 84565
   Exchange-mode: Aggressive
   Diffie-Hellman group: Group 1
   NAT traversal: Detected

   Extend authentication: Disabled
   Assigned IP address:
   Vendor ID index: 0xa1d
   Vendor ID sequence number: 0x0
```

# Display the IPsec SAs generated on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1435
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 830667426 (0x3182faa2)
      Connection ID: 90194313219
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: Y
      Status: Active
```

```
[Outbound ESP SAs]
  SPI: 3516214669 (0xd1952d8d)
  Connection ID: 64424509441
  Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
  SA duration (kilobytes/sec): 1843200/3600
  SA remaining duration (kilobytes/sec): 1843200/2313
  Max sent sequence-number:
  UDP encapsulation used for NAT traversal: Y
  Status: Active
```

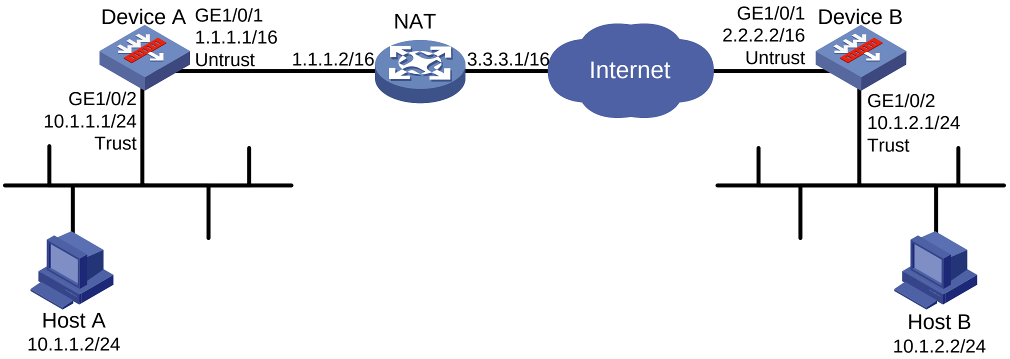
Example: Configuring an IKEv2-based IPsec NAT traversal tunnel for IPv4 packets between gateways

Network configuration

As shown in Figure 17, Device A is behind the NAT device. Configure an IKE-based IPsec tunnel between Device A and Device B to secure the communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKEv2 proposal and the default IKEv2 policy in IKEv2 negotiation to set up IPsec SAs.
- Configure the two devices to use the preshared key authentication method in IKEv2 negotiation.

Figure 17 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 1.1.1.2
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   ```

```
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

      ```
      [DeviceA-security-policy-ip] rule name untrust-trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] action pass
      [DeviceA-security-policy-ip-4-untrust-trust] quit
      [DeviceA-security-policy-ip] quit
      ```

5. Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule 0 permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set transform1
[DeviceA-ipsec-transform-set-transform1] protocol esp
[DeviceA-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceA-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceA-ipsec-transform-set-transform1] quit
```

7. Configure an IKEv2 keychain to specify the key information used for IKEv2 communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ikev2 keychain keychain1
[DeviceA-ikev2-keychain-keychain1] peer peer1
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 16
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
[DeviceA-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext 123
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceA-ikev2-keychain-keychain1] quit
```

8. Configure an IKEv2 profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] identity local fqdn www.devicea.com
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
[DeviceA-ikev2-profile-profile1] authentication-method local pre-share
[DeviceA-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceA-ikev2-profile-profile1] quit
```

9. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ipsec policy policy1 1 isakmp
[DeviceA-ipsec-policy-isakmp-policy1-1] remote-address 2.2.2.2
[DeviceA-ipsec-policy-isakmp-policy1-1] transform-set transform1
[DeviceA-ipsec-policy-isakmp-policy1-1] security acl 3101
[DeviceA-ipsec-policy-isakmp-policy1-1] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-policy1-1] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
[DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host B and Host A:

   # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
```

```
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

   # Configure IPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

   ```
   [DeviceA] acl advanced 3101
   [DeviceA-acl-ipv4-adv-3101] rule 0 permit ip source 10.1.2.0 0.0.0.255 destination
   10.1.1.0 0.0.0.255
   [DeviceA-acl-ipv4-adv-3101] quit
   ```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

   ```
   <DeviceB> system-view
   [DeviceB] ipsec transform-set transform1
   [DeviceB-ipsec-transform-set-transform1] protocol esp
   [DeviceB-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
   [DeviceB-ipsec-transform-set-transform1] esp authentication-algorithm md5
   [DeviceB-ipsec-transform-set-transform1] quit
   ```

7. Configure an IKEv2 keychain to specify the key information used by both sides of the IKE communication.

   ```
   [DeviceB]ikev2 keychain keychain1
   [DeviceB-ikev2-keychain-keychain1] peer peer1
   [DeviceB-ikev2-keychain-keychain1-peer-peer1] address 3.3.3.1 16
   [DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 3.3.3.1
   [DeviceB-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext 123
   [DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
   [DeviceB-ikev2-keychain-keychain1] quit
   ```

8. Configure an IKEv2 profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceB] ikev2 profile profile1
   [DeviceB-ikev2-profile-profile1] keychain keychain1
   [DeviceB-ikev2-profile-profile1] match remote identity fqdn www.devicea.com
   [DeviceB-ikev2-profile-profile1] authentication-method local pre-share
   [DeviceB-ikev2-profile-profile1] authentication-method remote pre-share
   [DeviceB-ikev2-profile-profile1] quit
   ```

9. Configure an IPsec policy template, which is used to create an IPsec policy.

   ```
   [DeviceB] ipsec policy-template template1 1
   [DeviceB-ipsec-policy-template-template1-1] remote-address 3.3.3.1
   [DeviceB-ipsec-policy-template-template1-1] security acl 3101
   [DeviceB-ipsec-policy-template-template1-1] transform-set transform1
   [DeviceB-ipsec-policy-template-template1-1] ikev2-profile profile1
   [DeviceB-ipsec-policy-template-template1-1] quit
   ```

10. Create an IKE-based IPsec policy entry by using IPsec policy template, so as to establish an IPsec tunnel to protect the specified data flow.

    ```
    [DeviceB] ipsec policy policy1 1 isakmp template template1
    ```

11. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/1
    ```

```
    [DeviceB-GigabitEthernet1/0/1] ipsec apply policy policy1
    [DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec-protected.

# Display the IKEv2 SA on Device A.

```
[DeviceA] display ikev2 sa
Tunnel ID   Local                       Remote                      Status
---------------------------------------------------------------------------
  1        1.1.1.1/4500                 2.2.2.2/4500                   EST
Status:
IN-NEGO: Negotiating, EST: Established, DEL:Deleting
 [DeviceA] display ikev2 sa verbose
  Tunnel ID: 45
  Local IP/Port: 1.1.1.1/4500
  Remote IP/Port: 2.2.2.2/4500
  Outside VRF: -
  Inside VRF: -
  Local SPI: 372228d699a33c63
  Remote SPI: 75c537621b4a7190

  Local ID type: ID_FQDN
  Local ID: www.devicea.com
  Remote ID type: ID_IPV4_ADDR
  Remote ID: 2.2.2.2

  Auth sign method: Pre-shared key
  Auth verify method: Pre-shared key
  Integrity algorithm: SHA1
  PRF algorithm: SHA1
  Encryption algorithm: AES-CBC-128

  Life duration: 86400 secs
  Remaining key duration: 86177 secs
  Diffie-Hellman group: MODP1536/Group5
  NAT traversal: Detected
  DPD: Interval 0 secs, retry interval 0 secs
  Transmitting entity: Initiator

  Local window: 1
  Remote window: 1
  Local request message ID: 2
  Remote request message ID: 0
  Local next message ID: 2
  Remote next message ID: 0
```

# Display the IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
```

```
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Path MTU: 1435
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.2.1.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 830667426 (0x3182faa2)
      Connection ID: 605590388736
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: Y
      Status: Active

    [Outbound ESP SAs]
      SPI: 3516214669 (0xd1952d8d)
      Connection ID: 227633266689
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max sent sequence-number:
      UDP encapsulation used for NAT traversal: Y
      Status: Active
```

Example: Configuring an IKE-based IPsec tunnel for IPv4 packets between a host and a gateway (remote extended authentication)

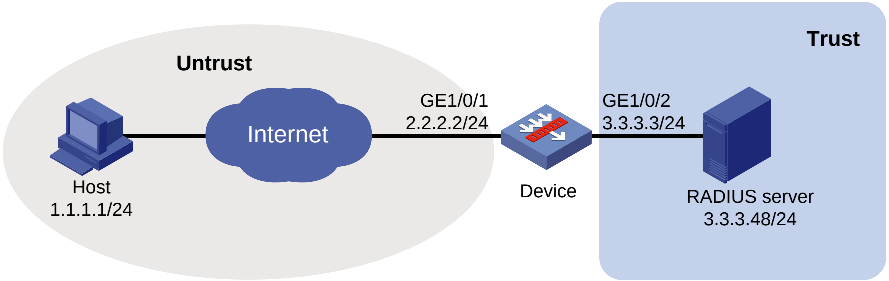
Network configuration

As shown in Figure 18, configure an IPsec tunnel to protect the traffic between the host and the device.

- Set up IPsec SAs through IKE negotiations.
- Configure the host and the device to use preshared key for authentication in the phase-1 IKE negotiation.
- Configure the device to use RADIUS to perform remote extended authentication on the host.

The RADIUS server is configured to allow user login with username **test** and password **123456TESTplat&!**.

Figure 18 Network diagram



Configuring the device

1. Assign IP addresses to interfaces.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
[Device] interface GigabitEthernet1/0/2
[Device-GigabitEthernet1/0/2] ip address 3.3.3.3 255.255.255.0
[Device-GigabitEthernet1/0/2] quit
```

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 2.2.2.3.

```
[Device] ip route-static 1.1.1.1 24 2.2.2.3
```

3. Add interfaces to security zones.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

4. Configure security policy rules to permit the traffic between the **Untrust** and **Local** security zones, so the host can access the device:

   # Configure a rule named **ipseclocalout1** to allow the device to send packets to the host.

```
[Device] security-policy ip
```

```
[Device-security-policy-ip] rule name ipseclocalout1
[Device-security-policy-ip-1-ipseclocalout1] source-zone local
[Device-security-policy-ip-1-ipseclocalout1] destination-zone untrust
[Device-security-policy-ip-1-ipseclocalout1] source-ip-host 2.2.2.2
[Device-security-policy-ip-1-ipseclocalout1] destination-ip-host 1.1.1.1
[Device-security-policy-ip-1-ipseclocalout1] action pass
[Device-security-policy-ip-1-ipseclocalout1] quit
```

# Configure a rule named **ipseclocalin1** to allow the device to receive the packets sent from the host.

```
[Device-security-policy-ip] rule name ipseclocalin1
[Device-security-policy-ip-2-ipseclocalin1] source-zone untrust
[Device-security-policy-ip-2-ipseclocalin1] destination-zone local
[Device-security-policy-ip-2-ipseclocalin1] source-ip-host 1.1.1.1
[Device-security-policy-ip-2-ipseclocalin1] destination-ip-host 2.2.2.2
[Device-security-policy-ip-2-ipseclocalin1] action pass
[Device-security-policy-ip-2-ipseclocalin1] quit
```

# Configure a rule named **ipseclocalout2** to allow the device to send packets to the RADIUS server.

```
[Device-security-policy-ip] rule name ipseclocalout2
[Device-security-policy-ip-3-ipseclocalout2] source-zone local
[Device-security-policy-ip-3-ipseclocalout2] destination-zone trust
[Device-security-policy-ip-3-ipseclocalout2] source-ip-host 3.3.3.3
[Device-security-policy-ip-3-ipseclocalout2] destination-ip-host 3.3.3.48
[Device-security-policy-ip-3-ipseclocalout2] action pass
[Device-security-policy-ip-3-ipseclocalout2] quit
```

# Configure a rule named **ipseclocalin2** to allow the device to receive the packets sent from the RADIUS server.

```
[Device-security-policy-ip] rule name ipseclocalin2
[Device-security-policy-ip-4-ipseclocalin2] source-zone trust
[Device-security-policy-ip-4-ipseclocalin2] destination-zone local
[Device-security-policy-ip-4-ipseclocalin2] source-ip-host 3.3.3.48
[Device-security-policy-ip-4-ipseclocalin2] destination-ip-host 3.3.3.3
[Device-security-policy-ip-4-ipseclocalin2] action pass
[Device-security-policy-ip-4-ipseclocalin2] quit
[Device-security-policy-ip] quit
```

5. Configure a RADIUS scheme:

# Create a RADIUS scheme named **ike-scheme**.

```
[Device] radius scheme ike-scheme
```

# Specify the IP address and service port of the primary RADIUS authentication server.

```
[Device-radius-ike-scheme] primary authentication 3.3.3.48 1645
```

# Set the shared key for secure RADIUS authentication communication.

```
[Device-radius-ike-scheme] key authentication simple abc
```

# Configure the device to send the username without the ISP domain name to the RADIUS server. (The configuration varies with the RADIUS server's requirements for username.)

```
[Device-radius-ike-scheme] user-name-format without-domain
[Device-radius-ike-scheme] quit
```

6. Configure an ISP domain:

Create an ISP domain named **ike** and specify the RADIUS scheme used for authenticating the IKE users.

```
[Device] domain ike
[Device-isp-ike] authentication ike radius-scheme ike-scheme
[Device-isp-ike] quit
```

7. Configure an IPv4 advanced ACL to identify the packets to be protected.

```
[Device] acl advanced 3101
[Device-acl-ipv4-adv-3101] rule permit ip source 2.2.2.2 0.0.0.0 destination 1.1.1.1
0.0.0.0
[Device-acl-ipv4-adv-3101] quit
```

8. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[Device] ipsec transform-set tran1
[Device-ipsec-transform-set-tran1] encapsulation-mode transport
[Device-ipsec-transform-set-tran1] protocol esp
[Device-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[Device-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[Device-ipsec-transform-set-tran1] quit
```

9. Configure an IKE keychain to specify the key information used for IKE communication:

   The preshared key used by both sides of the communication must be the same.

```
[Device] ike keychain keychain1
[Device-ike-keychain-keychain1] pre-shared-key address 1.1.1.1 255.255.255.255 key
simple 123456TESTplat&!
[Device-ike-keychain-keychain1] quit
```

10. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[Device] ike profile profile1
[Device-ike-profile-profile1] keychain keychain1
[Device-ike-profile-profile1] local-identity address 2.2.2.2
[Device-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.255.255
[Device-ike-profile-profile1] client-authentication xauth
[Device-ike-profile-profile1] quit
```

11. Configure an IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[Device] ipsec policy map1 10 isakmp
[Device-ipsec-policy-isakmp-map1-10] remote-address 1.1.1.1
[Device-ipsec-policy-isakmp-map1-10] security acl 3101
[Device-ipsec-policy-isakmp-map1-10] transform-set tran1
[Device-ipsec-policy-isakmp-map1-10] ike-profile profile1
[Device-ipsec-policy-isakmp-map1-10] quit
```

12. Apply the IPsec policy to GigabitEthernet 1/0/1.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ipsec apply policy map1
[Device-GigabitEthernet1/0/1] quit
```

Configuring the host

Perform the following tasks on the host and make sure the configuration matches that on the device:

- Specify the IP address of the remote security gateway.
- Set the preshared key used for IKE negotiation.

- Configure the username and password for IKE extended authentication.
- Specify the security protocol, encryption algorithm, and authentication algorithm.
- Configure IKE negotiation parameters.
- Configure the local ID and remote ID.

(Details not shown.)

Verifying the configuration

# Initiate a connection from the host (1.1.1.1) to the device (2.2.2.2) to trigger IKE negotiation. (Details not shown.)

# On the device, verify that an IKE SA to the peer 1.1.1.1 is established and that extended authentication is enabled for remote users.

```
[Device] display ike sa verbose remote-address 1.1.1.1
   -----------------------------------------------
   Connection ID: 18
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Initiator
   Initiator cookie: 1bcf453f0a217259
   Responder cookie: 5e32a74dfa66a0a4
   -----------------------------------------------
   Local IP/port: 2.2.2.2/500
   Local ID type: IPV4_ADDR
   Local ID: 2.2.2.2

   Remote IP/port: 1.1.1.1/500
   Remote ID type: IPV4_ADDR
   Remote ID: 1.1.1.1

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: SHA1
   Encryption-algorithm: DES-CBC

   Life duration(sec): 86400
   Remaining key duration(sec): 84565
   Exchange-mode: Aggressive
   Diffie-Hellman group: Group 1
   NAT traversal: Detected

   Extend authentication: Enabled
   Assigned IP address:
   Vendor ID index: 0xa1d
   Vendor ID sequence number: 0x0
```

# On the host, enter the correct username and password for extended authentication. After the authentication succeeds, the IPsec tunnel will be established. (Details not shown.)

# Verify that IPsec SAs have been established on the device.

```
[Device] display ipsec sa
```

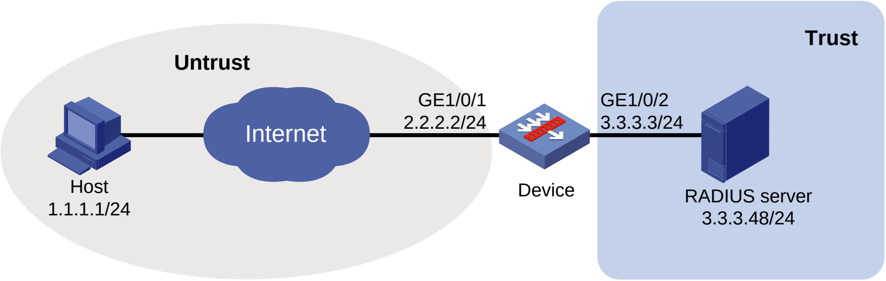
Example: Configuring an IKE-based IPsec tunnel for IPv4 packets between a host and a gateway (local extended authentication and address pool authorization)

Network configuration

As shown in Figure 19, configure an IPsec tunnel to protect the traffic between the host and the server.

- Set up IPsec SAs through IKE negotiations.
- Configure the host and the device to use preshared key for authentication in the phase-1 IKE negotiation.
- Configure the device to use AAA to perform local extended authentication on the host and assign an IPv4 address to the host.

Figure 19 Network diagram



Restrictions and guidelines

Make sure the host, device, and server can reach each other.

Configure a local user account on the device to provide identity authentication for the host. In this example, the account uses username **test** and password **123456TESTplat&!**.

Configuring the device

1. Assign IP addresses to interfaces.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
[Device] interface GigabitEthernet1/0/2
[Device-GigabitEthernet1/0/2] ip address 3.3.3.3 255.255.255.0
[Device-GigabitEthernet1/0/2] quit
```

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 2.2.2.3.

```
[Device] ip route-static 1.1.1.1 24 2.2.2.3
```

3. Add interfaces to security zones.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

4. Configure security policy rules to permit the traffic between the **Untrust** and **Local** security zones, so the host can access the device:

   # Configure a rule named **ipseclocalout1** to allow the device to send packets to the host.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name ipseclocalout1
   [Device-security-policy-ip-1-ipseclocalout1] source-zone local
   [Device-security-policy-ip-1-ipseclocalout1] destination-zone untrust
   [Device-security-policy-ip-1-ipseclocalout1] source-ip-host 2.2.2.2
   [Device-security-policy-ip-1-ipseclocalout1] destination-ip-host 1.1.1.1
   [Device-security-policy-ip-1-ipseclocalout1] action pass
   [Device-security-policy-ip-1-ipseclocalout1] quit
   ```

   # Configure a rule named **ipseclocalin1** to allow the device to receive the packets sent from the host.

   ```
   [Device-security-policy-ip] rule name ipseclocalin1
   [Device-security-policy-ip-2-ipseclocalin1] source-zone untrust
   [Device-security-policy-ip-2-ipseclocalin1] destination-zone local
   [Device-security-policy-ip-2-ipseclocalin1] source-ip-host 1.1.1.1
   [Device-security-policy-ip-2-ipseclocalin1] destination-ip-host 2.2.2.2
   [Device-security-policy-ip-2-ipseclocalin1] action pass
   [Device-security-policy-ip-2-ipseclocalin1] quit
   ```

   # Configure a rule named **ipseclocalout2** to allow the device to send packets to the RADIUS server.

   ```
   [Device-security-policy-ip] rule name ipseclocalout2
   [Device-security-policy-ip-3-ipseclocalout2] source-zone local
   [Device-security-policy-ip-3-ipseclocalout2] destination-zone trust
   [Device-security-policy-ip-3-ipseclocalout2] source-ip-host 3.3.3.3
   [Device-security-policy-ip-3-ipseclocalout2] destination-ip-host 3.3.3.48
   [Device-security-policy-ip-3-ipseclocalout2] action pass
   [Device-security-policy-ip-3-ipseclocalout2] quit
   ```

   # Configure a rule named **ipseclocalin2** to allow the device to receive the packets sent from the RADIUS server.

   ```
   [Device-security-policy-ip] rule name ipseclocalin2
   [Device-security-policy-ip-4-ipseclocalin2] source-zone trust
   [Device-security-policy-ip-4-ipseclocalin2] destination-zone local
   [Device-security-policy-ip-4-ipseclocalin2] source-ip-host 3.3.3.48
   [Device-security-policy-ip-4-ipseclocalin2] destination-ip-host 3.3.3.3
   [Device-security-policy-ip-4-ipseclocalin2] action pass
   [Device-security-policy-ip-4-ipseclocalin2] quit
   [Device-security-policy-ip] quit
   ```

5. Configure an ISP domain:

   # Create an ISP domain named **dm**.

   ```
   [Device] domain dm
   ```

   # Configure the device to perform IKE local authentication.

   ```
   [Device-isp-dm] authentication ike local
   ```

   # Configure the device to perform IKE local authorization.

   ```
   [Device-isp-dm] authorization ike local
   [Device-isp-dm] quit
   ```

6. Create an IKE IPv4 address pool named **pool** with the address range 20.1.1.1 to 20.1.1.20.

```
[Device] ike address-group pool 20.1.1.1 20.1.1.20
```

7. Configure local users:

   # Add a network user named **ike**.

   ```
   [Device] local-user ike class network
   ```

   # Authorize user **ike** to use the IKE service. Specify IPv4 address pool **pool** as the authorized IPv4 address pool for user **ike**.

   ```
   [Device-luser-network-ike] service-type ike
   [Device-luser-network-ike] authorization-attribute ip-pool pool
   [Device-luser-network-ike] quit
   ```

   # Add a network user named **test**.

   ```
   [Device] local-user test class network
   ```

   # Authorize user **test** to use the IKE service, and configure a password for user **test**.

   ```
   [Device-luser-network-test] service-type ike
   [Device-luser-network-test] password simple 123456TESTplat&!
   [Device-luser-network-test] quit
   ```

8. Configure an IKE keychain to specify the key information used for IKE communication:

   The preshared key used by both sides of the communication must be the same.

   ```
   [Device] ike keychain keychain1
   [Device-ike-keychain-keychain1] pre-shared-key address 1.1.1.1 255.255.255.255 key
   simple 123456TESTplat&!
   [Device-ike-keychain-keychain1] quit
   ```

9. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [Device] ike profile profile1
   [Device-ike-profile-profile1] keychain keychain1
   [Device-ike-profile-profile1] local-identity address 2.2.2.2
   [Device-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.255.255
   [Device-ike-profile-profile1] client-authentication xauth
   [Device-ike-profile-profile1] aaa authorization domain dm username ike
   [Device-ike-profile-profile1] quit
   ```

10. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

    The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

    ```
    [Device] ipsec transform-set tran1
    [Device-ipsec-transform-set-tran1] encapsulation-mode transport
    [Device-ipsec-transform-set-tran1] protocol esp
    [Device-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-256
    [Device-ipsec-transform-set-tran1] esp authentication-algorithm sha1
    [Device-ipsec-transform-set-tran1] quit
    ```

11. Configure an IPsec policy template, which is used to create an IPsec policy.

    ```
    [Device] ipsec policy-template pt 1
    [Device-ipsec-policy-template-pt-1] transform-set tran1
    [Device-ipsec-policy-template-pt-1] ike-profile profile1
    [Device-ipsec-policy-template-pt-1] reverse-route dynamic
    [Device-ipsec-policy-template-pt-1] quit
    ```

12. Use the IPsec policy template to create an IKE-based IPsec policy entry.

    ```
    [Device] ipsec policy map1 1 isakmp template pt
    ```

13. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect the traffic on the interface.

    ```
    [Device] interface gigabitethernet 1/0/1
    ```

```
    [Device-GigabitEthernet1/0/1] ipsec apply policy map1
    [Device-GigabitEthernet1/0/1] quit
```

Configuring the host

Perform the following tasks on the host and make sure the configuration matches that on the device:

- Specify the IP address of the remote security gateway.
- Set the preshared key used for IKE negotiation.
- Configure the username and password for IKE client authentication.
- Specify the security protocol, encryption algorithm, and authentication algorithm.
- Configure IKE negotiation parameters.
- Configure the local ID and remote ID.

(Details not shown.)

Verifying the configuration

# Initiate a connection from the host (1.1.1.1) to the server (3.3.3.48) to trigger IKE negotiation. (Details not shown.)

# On the device, verify that an IKE SA to peer 1.1.1.1 is established and client authentication is enabled.

```
[Device] display ike sa verbose remote-address 1.1.1.1
   -----------------------------------------------
   Connection ID: 18
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Responder
   -----------------------------------------------
   Local IP/port: 2.2.2.2/500
   Local ID type: IPV4_ADDR
   Local ID: 2.2.2.2

   Remote IP/port: 1.1.1.1/500
   Remote ID type: IPV4_ADDR
   Remote ID: 1.1.1.1

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: SHA1
   Encryption-algorithm: 3DES-CBC

   Life duration(sec): 86400
   Remaining key duration(sec): 84565
   Exchange-mode: Main
   Diffie-Hellman group: Group 2
   NAT traversal: Detected

   Extend authentication: Enabled
   Assigned IP address: 20.1.1.2
```

# On the host, enter the correct username and password for client authentication. After the authentication succeeds, the IPsec tunnel will be established. (Details not shown.)

Verify that IPsec SAs are established on the device.

```
<Device> display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------
  -----------------------------
  IPsec policy: map1
  Sequence number: 1
  Mode: Template
  -----------------------------
    Tunnel id: 2
    Encapsulation mode: transport
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1427
    Tunnel:
        local  address: 2.2.2.2
        remote address: 1.1.1.1
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 20.1.1.2/255.255.255.255  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2374047012 (0x8d811524)
      Transform set: ESP-ENCRYPT-AES-CBC-256 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843198/3259
      Max received sequence-number: 24
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 146589619 (0x08bcc7b3)
      Transform set: ESP-ENCRYPT-AES-CBC-256 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3259
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1839568/3164
      Max sent sequence-number: 2793
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

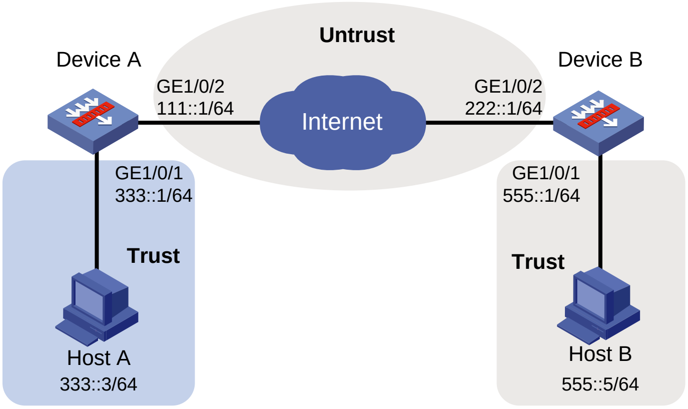
Example: Configuring an IKE-based IPsec tunnel for IPv6 packets between gateways

Network configuration

As shown in Figure 20, establish an IPsec tunnel between Device A and Device B to protect data flows between subnet 333::/64 and subnet 555::/64. Configure the IPsec tunnel as follows:

- Specify the encapsulation mode as tunnel, the security protocol as ESP, the encryption algorithm as 128-bit AES, and the authentication algorithm as HMAC-SHA1.
- Set up SAs through IKE negotiation.

Figure 20 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ipv6 address 333::1/64
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 111::2.

   ```
   [DeviceA] ipv6 route-static 555::0 64 111::2
   [DeviceA] ipv6 route-static 222::0 64 111::2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name ipseclocalout
[DeviceA-security-policy-ipv6-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ipv6-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-ipseclocalout] source-ip-host 111::1
[DeviceA-security-policy-ipv6-1-ipseclocalout] destination-ip-host 222::1
[DeviceA-security-policy-ipv6-1-ipseclocalout] action pass
[DeviceA-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ipv6] rule name ipseclocalin
[DeviceA-security-policy-ipv6-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-ipseclocalin] source-ip-host 222::1
[DeviceA-security-policy-ipv6-2-ipseclocalin] destination-ip-host 111::1
[DeviceA-security-policy-ipv6-2-ipseclocalin] action pass
[DeviceA-security-policy-ipv6-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-ip-subnet 333::1 64
[DeviceA-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 555::1 64
[DeviceA-security-policy-ipv6-3-trust-untrust] action pass
[DeviceA-security-policy-ipv6-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-ip-subnet 555::1 64
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-ip-subnet 333::1 64
[DeviceA-security-policy-ipv6-4-untrust-trust] action pass
[DeviceA-security-policy-ipv6-4-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

5. Define the data flows to be protected:

# Configure an IPv6 advanced ACL to identify data flows from subnet 333::/64 to subnet 555::/64.

```
[DeviceA] acl ipv6 advanced 3101
[DeviceA-acl-ipv6-adv-3101] rule permit ipv6 source 333::0 64 destination 555::0 64
[DeviceA-acl-ipv6-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
```

```
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. **Configure an IKE keychain to specify the key information used for IKE communication.**

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceA] ike keychain keychain1
   [DeviceA-ike-keychain-keychain1] pre-shared-key address ipv6 222::1 64 key simple
   123456TESTplat&!
   [DeviceA-ike-keychain-keychain1] quit
   ```

8. **Configure an IKE profile to specify the security parameters used for setting up IKE SAs.**

   ```
   [DeviceA] ike profile profile1
   [DeviceA-ike-profile-profile1] keychain keychain1
   [DeviceA-ike-profile-profile1] match remote identity address ipv6 222::1 64
   [DeviceA-ike-profile-profile1] quit
   ```

9. **Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:**

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceA] ipsec ipv6-policy map1 10 isakmp
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] security acl ipv6 3101
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] transform-set tran1
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] local-address ipv6 111::1
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] remote-address ipv6 222::1
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] ike-profile profile1
   [DeviceA-ipsec-ipv6-policy-isakmp-map1-10] quit
   ```

10. **Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.**

   ```
   [DeviceA] interface gigabitethernet 1/0/2
   [DeviceA-GigabitEthernet1/0/2] ipsec apply ipv6-policy map1
   [DeviceA-GigabitEthernet1/0/2] quit
   ```

Configuring Device B

1. **Assign IP addresses to interfaces:**

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ipv6 address 555::1/64
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. **Configure settings for routing.**

   This example configures static routes, and the next hop in the routes is 222::2.

   ```
   [DeviceB] ipv6 route-static 333::0 64 222::2
   [DeviceB] ipv6 route-static 111::0 64 222::2
   ```

3. **Add interfaces to security zones.**

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Untrust] quit
   ```

4. **Configure a security policy:**

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name ipseclocalout
[DeviceB-security-policy-ipv6-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ipv6-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-ipseclocalout] source-ip-host 222::1
[DeviceB-security-policy-ipv6-1-ipseclocalout] destination-ip-host 111::1
[DeviceB-security-policy-ipv6-1-ipseclocalout] action pass
[DeviceB-security-policy-ipv6-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ipv6] rule name ipseclocalin
[DeviceB-security-policy-ipv6-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ipv6-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ipv6-2-ipseclocalin] source-ip-host 111::1
[DeviceB-security-policy-ipv6-2-ipseclocalin] destination-ip-host 222::1
[DeviceB-security-policy-ipv6-2-ipseclocalin] action pass
[DeviceA-security-policy-ipv6-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

   # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-ip-subnet 333::1 64
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 555::1 64
[DeviceB-security-policy-ipv6-3-trust-untrust] action pass
[DeviceB-security-policy-ipv6-3-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ipv6] rule name untrust-trust
[DeviceB-security-policy-ipv6-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ipv6-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ipv6-4-untrust-trust] source-ip-subnet 555::1 64
[DeviceB-security-policy-ipv6-4-untrust-trust] destination-ip-subnet 333::1 64
[DeviceB-security-policy-ipv6-4-untrust-trust] action pass
[DeviceB-security-policy-ipv6-4-untrust-trust] quit
[DeviceB-security-policy-ipv6] quit
```

5. Define the data flows to be protected:

   # Configure an IPv6 advanced ACL to identify data flows from subnet 555::/64 to subnet 333::/64.

```
[DeviceB] acl ipv6 advanced 3101
[DeviceB-acl-ipv6-adv-3101] rule permit ipv6 source 555::/64 destination 333::/64
[DeviceB-acl-ipv6-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceB] ike keychain keychain1
   [DeviceB-ike-keychain-keychain1] pre-shared-key address ipv6 111::1 64 key simple
   123456TESTplat&!
   [DeviceB-ike-keychain-keychain1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceB] ike profile profile1
   [DeviceB-ike-profile-profile1] keychain keychain1
   [DeviceB-ike-profile-profile1] match remote identity address ipv6 111::1 64
   [DeviceB-ike-profile-profile1] quit
   ```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceB] ipsec ipv6-policy use1 10 isakmp
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] security acl ipv6 3101
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] transform-set tran1
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] local-address ipv6 222::1
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] remote-address ipv6 111::1
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] ike-profile profile1
   [DeviceB-ipsec-ipv6-policy-isakmp-use1-10] quit
   ```

10. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/2
    [DeviceB-GigabitEthernet1/0/2] ipsec apply ipv6-policy use1
    [DeviceB-GigabitEthernet1/0/2] quit
    ```

Verifying the configuration

# Initiate a connection from subnet 333::/64 to subnet 555::/64 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, the traffic between the two subnets is IPsec-protected.

# Display IPsec SAs on Device A and Device B. This example uses Device A to verify the configuration.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/2
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
```

```
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1423
    Tunnel:
        local  address: 111::1
        remote address: 222::1
    Flow:
    sour addr: 111::1/0      port: 0  protocol: ipv6
    dest addr: 222::1/0      port: 0  protocol: ipv6

    [Inbound ESP SAs]
      SPI: 3769702703 (0xe0b1192f)
      Connection ID: 1
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 3000/28800
      SA remaining duration (kilobytes/sec): 2300/797
      Max received sequence-number: 1
      Anti-replay check enable: N
      Anti-replay window size:
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 3840956402 (0xe4f057f2)
      Connection ID: 2
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 3000/28800
      SA remaining duration (kilobytes/sec): 2312/797
      Max sent sequence-number: 1
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

Example: Configuring IPsec tunnels for IPv4 packets between headquarters and branches (IPsec policy template mode)

Network configuration

The branches of an enterprise access the headquarters through IPsec VPN. Device A is the headquarters gateway. Device B and Device C are the branch gateways.

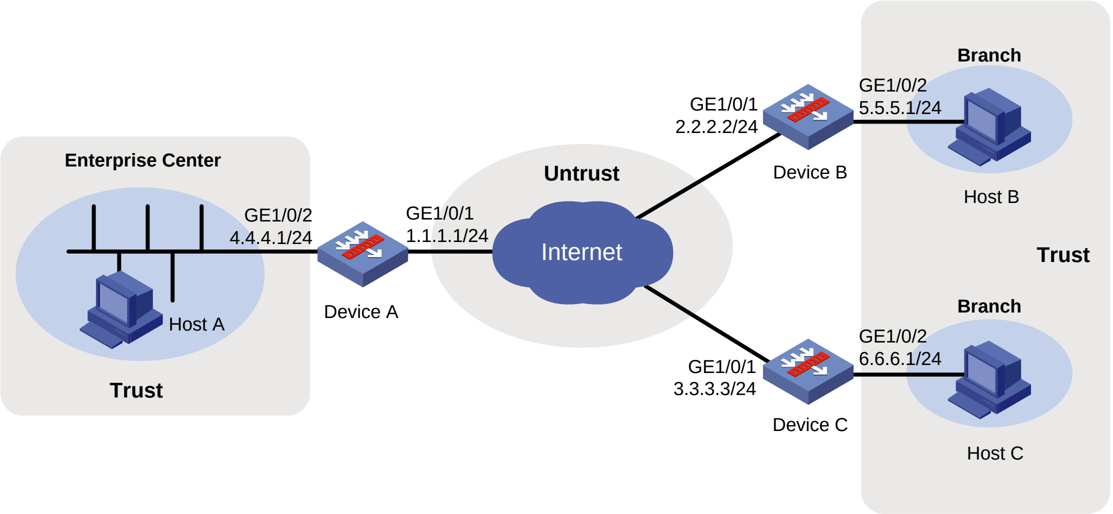
Establish an IPsec tunnel between the headquarters gateway and each branch gateway to protect the data between the headquarters network (4.4.4.0/24) and the branch networks (5.5.5.0/24 and 6.6.6.0/24)

Configure the headquarters gateway Device A to use an IKE-based IPsec policy template and the branch gateways Device B and Device C to use an IKE-based IPsec policy to establish the IPsec tunnels.

Use the ESP security protocol, DES encryption algorithm, and HMAC-SHA-1-96 authentication algorithm to establish IPsec SAs.

Use the preshared key authentication mode, 3DES encryption algorithm, and HMAC-SHA1 authentication algorithm for IKE negotiation.

Figure 21 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

```
[DeviceA] ip route-static 2.2.2.2 24 1.1.1.2
[DeviceA] ip route-static 3.3.3.3 24 1.1.1.2
[DeviceA] ip route-static 5.5.5.0 255.255.255.0 1.1.1.2
[DeviceA] ip route-static 6.6.6.0 255.255.255.0 1.1.1.2
```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout1** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
```

```
[DeviceA-security-policy-ip] rule name ipseclocalout1
[DeviceA-security-policy-ip-1-ipseclocalout1] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout1] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout1] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout1] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout1] action pass
[DeviceA-security-policy-ip-1-ipseclocalout1] quit
```

# Configure a rule named **ipseclocalin1** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin1
[DeviceA-security-policy-ip-2-ipseclocalin1] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin1] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin1] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin1] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin1] action pass
[DeviceA-security-policy-ip-2-ipseclocalin1] quit
```

# Configure a rule named **ipseclocalout2** to allow Device A to send IPsec negotiation packets to Device C.

```
[DeviceA-security-policy-ip] rule name ipseclocalout2
[DeviceA-security-policy-ip-3-ipseclocalout2] source-zone local
[DeviceA-security-policy-ip-3-ipseclocalout2] destination-zone untrust
[DeviceA-security-policy-ip-3-ipseclocalout2] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-3-ipseclocalout2] destination-ip-host 3.3.3.3
[DeviceA-security-policy-ip-3-ipseclocalout2] action pass
[DeviceA-security-policy-ip-3-ipseclocalout2] quit
```

# Configure a rule named **ipseclocalin2** to allow Device A to receive the IPsec negotiation packets sent from Device C.

```
[DeviceA-security-policy-ip] rule name ipseclocalin2
[DeviceA-security-policy-ip-4-ipseclocalin2] source-zone untrust
[DeviceA-security-policy-ip-4-ipseclocalin2] destination-zone local
[DeviceA-security-policy-ip-4-ipseclocalin2] source-ip-host 3.3.3.3
[DeviceA-security-policy-ip-4-ipseclocalin2] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-4-ipseclocalin2] action pass
[DeviceA-security-policy-ip-4-ipseclocalin2] quit
```

b. Configure rules to permit the traffic between Host A and Host B or Host C:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-5-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-5-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-5-trust-untrust] source-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-5-trust-untrust] destination-ip-subnet 5.5.5.0 24
[DeviceA-security-policy-ip-5-trust-untrust] action pass
[DeviceA-security-policy-ip-5-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-6-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-6-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-6-untrust-trust] source-ip-subnet 5.5.5.0 24
```

```
[DeviceA-security-policy-ip-6-untrust-trust] destination-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-6-untrust-trust] action pass
[DeviceA-security-policy-ip-6-untrust-trust] quit
```

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host C.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-7-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-7-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-7-trust-untrust] source-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-7-trust-untrust] destination-ip-subnet 6.6.6.0 24
[DeviceA-security-policy-ip-7-trust-untrust] action pass
[DeviceA-security-policy-ip-7-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host C to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-8-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-8-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-8-untrust-trust] source-ip-subnet 6.6.6.0 24
[DeviceA-security-policy-ip-8-untrust-trust] destination-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-8-untrust-trust] action pass
[DeviceA-security-policy-ip-8-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms. The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

6. Configure IKE keychains to specify the keys information used for IKE communication between peers.

# Create an IKE keychain named **key1** to specify the preshared key as **123** for communication with peer 2.2.2.2.

```
[DeviceA] ike keychain key1
[DeviceA-ike-keychain-key1] pre-shared-key address 2.2.2.2 key simple 123
[DeviceA-ike-keychain-key1] quit
```

# Create an IKE keychain named **key2** to specify the preshared key as **456** for communication with peer 3.3.3.3.

```
[DeviceA] ike keychain key2
[DeviceA-ike-keychain-key2] pre-shared-key address 3.3.3.3 key simple 456
[DeviceA-ike-keychain-key2] quit
```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] keychain key1
[DeviceA-ike-profile-profile1] keychain key2
[DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.255.0
[DeviceA-ike-profile-profile1] match remote identity address 3.3.3.3 255.255.255.0
[DeviceA-ike-profile-profile1] quit
```

8. Configure an IPsec policy template, which is used to create an IPsec policy:

   # Create an IPsec policy template named **temp1**. Specify the IPsec transform set **tran1** and IKE profile **profile1** for the template.

   ```
   [DeviceA] ipsec policy-template temp1 1
   [DeviceA-ipsec-policy-template-temp1-1] transform-set tran1
   [DeviceA-ipsec-policy-template-temp1-1] ike-profile profile1
   ```

9. Create an IKE-based IPsec policy entry by using IPsec policy template **temp1**, so as to establish the IPsec tunnel to protect data.

   ```
   [DeviceA] ipsec policy map1 10 isakmp template temp1
   ```

10. Configure an IKE proposal to specify the security parameters used for IKE negotiation.

    # Create an IKE proposal named **1**, which uses the 3DES encryption algorithm, HMAC-SHA1 authentication algorithm, and preshared key authentication method.

    ```
    [DeviceA] ike proposal 1
    [DeviceA-ike-proposal-1] encryption-algorithm 3des-cbc
    [DeviceA-ike-proposal-1] authentication-algorithm sha
    [DeviceA-ike-proposal-1] authentication-method pre-share
    [DeviceA-ike-proposal-1] quit
    ```

11. Apply the IPsec policy to interface GigabitEthernet 1/0/1 to protect traffic on the interface.

    ```
    [DeviceA] interface gigabitethernet 1/0/1
    [DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
    [DeviceA-GigabitEthernet1/0/1] quit
    ```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3.

   ```
   [DeviceB] ip route-static 4.4.4.0 24 2.2.2.3
   [DeviceB] ip route-static 1.1.1.1 24 2.2.2.3
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      ```

```
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 5.5.5.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 4.4.4.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 4.4.4.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 5.5.5.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure an IPv4 advanced ACL to identify data flows from subnet 5.5.5.0/24 to subnet 4.4.4.0/24.

```
[DeviceB] acl advanced 3000
[DeviceB-acl-ipv4-adv-3000] rule permit ip source 5.5.5.0 0.0.0.255 destination
4.4.4.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3000] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms. The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   # Create an IKE keychain named **key1** to specify the preshared key as **123** for communication with peer 1.1.1.1.

   ```
   [DeviceB] ike keychain key1
   [DeviceB-ike-keychain-key1] pre-shared-key address 1.1.1.1 key simple 123
   [DeviceB-ike-keychain-key1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceB] ike profile profile1
   [DeviceB-ike-profile-profile1] keychain key1
   [DeviceB-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.255.0
   [DeviceB-ike-profile-profile1] quit
   ```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceB] ipsec policy map1 10 isakmp
   [DeviceB-ipsec-policy-isakmp-map1-10] transform-set tran1
   [DeviceB-ipsec-policy-isakmp-map1-10] security acl 3000
   [DeviceB-ipsec-policy-isakmp-map1-10] local-address 2.2.2.2
   [DeviceB-ipsec-policy-isakmp-map1-10] remote-address 1.1.1.1
   [DeviceB-ipsec-policy-isakmp-map1-10] ike-profile profile1
   [DeviceB-ipsec-policy-isakmp-map1-10] quit
   ```

10. Configure an IKE proposal to specify the security parameters used for IKE negotiation.

    # Create an IKE proposal named **1**, which uses the 3DES encryption algorithm, HMAC-SHA1 authentication algorithm, and preshared key authentication method.

    ```
    [DeviceB] ike proposal 1
    [DeviceB-ike-proposal-1] encryption-algorithm 3des-cbc
    [DeviceB-ike-proposal-1] authentication-algorithm sha
    [DeviceB-ike-proposal-1] authentication-method pre-share
    [DeviceB-ike-proposal-1] quit
    ```

11. Apply the IPsec policy to interface GigabitEthernet 1/0/1 to protect traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/1
    [DeviceB-GigabitEthernet1/0/1] ipsec apply policy map1
    [DeviceB-GigabitEthernet1/0/1] quit
    ```

Configuring Device C

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceC> system-view
   [DeviceC] interface gigabitethernet 1/0/1
   [DeviceC-GigabitEthernet1/0/1] ip address 3.3.3.3 255.255.255.0
   [DeviceC-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 3.3.3.4

   ```
   [DeviceC] ip route-static 4.4.4.0 24 3.3.3.4
   [DeviceC] ip route-static 1.1.1.1 24 3.3.3.4
   ```

3. Add interfaces to security zones.

   ```
   [DeviceC] security-zone name untrust
   ```

```
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceC-security-zone-Untrust] quit
[DeviceC] security-zone name trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device C to send IPsec negotiation packets to Device A.

      ```
      [DeviceC] security-policy ip
      [DeviceC-security-policy-ip] rule name ipseclocalout
      [DeviceC-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceC-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceC-security-policy-ip-1-ipseclocalout] source-ip-host 3.3.3.3
      [DeviceC-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceC-security-policy-ip-1-ipseclocalout] action pass
      [DeviceC-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device C to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceC-security-policy-ip] rule name ipseclocalin
      [DeviceC-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceC-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceC-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceC-security-policy-ip-2-ipseclocalin] destination-ip-host 3.3.3.3
      [DeviceC-security-policy-ip-2-ipseclocalin] action pass
      [DeviceC-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit traffic between Host C and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host C to Host A.

      ```
      [DeviceC-security-policy-ip] rule name trust-untrust
      [DeviceC-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceC-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceC-security-policy-ip-3-trust-untrust] source-ip-subnet 6.6.6.0 24
      [DeviceC-security-policy-ip-3-trust-untrust] destination-ip-subnet 4.4.4.0 24
      [DeviceC-security-policy-ip-3-trust-untrust] action pass
      [DeviceC-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host A to Host C.

      ```
      [DeviceC-security-policy-ip] rule name untrust-trust
      [DeviceC-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceC-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceC-security-policy-ip-4-untrust-trust] source-ip-subnet 4.4.4.0 24
      [DeviceC-security-policy-ip-4-untrust-trust] destination-ip-subnet 6.6.6.0 24
      [DeviceC-security-policy-ip-4-untrust-trust] action pass
      [DeviceC-security-policy-ip-4-untrust-trust] quit
      [DeviceC-security-policy-ip] quit
      ```

5. Define the data flows to be protected:

   # Configure an IPv4 advanced ACL to identify data flows from subnet 6.6.6.0/24 to subnet 4.4.4.0/24.

```
[DeviceC] acl advanced 3000
[DeviceC-acl-ipv4-adv-3000] rule permit ip source 6.6.6.0 0.0.0.255 destination 
4.4.4.0 0.0.0.255
[DeviceC-acl-ipv4-adv-3000] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms. The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceC] ipsec transform-set tran1
[DeviceC-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceC-ipsec-transform-set-tran1] protocol esp
[DeviceC-ipsec-transform-set-tran1] esp encryption-algorithm des
[DeviceC-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceC-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   # Create an IKE keychain named **key1** to specify the preshared key as **456** for communication with peer 1.1.1.1.

```
[DeviceC] ike keychain key1
[DeviceC-ike-keychain-key1] pre-shared-key address 1.1.1.1 key simple 456
[DeviceC-ike-keychain-key1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceC] ike profile profile1
[DeviceC-ike-profile-profile1] keychain key1
[DeviceC-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.255.0
[DeviceC-ike-profile-profile1] quit
```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

```
[DeviceC] ipsec policy map1 10 isakmp
[DeviceC-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceC-ipsec-policy-isakmp-map1-10] security acl 3000
[DeviceC-ipsec-policy-isakmp-map1-10] local-address 3.3.3.3
[DeviceC-ipsec-policy-isakmp-map1-10] remote-address 1.1.1.1
[DeviceC-ipsec-policy-isakmp-map1-10] ike-profile profile1
[DeviceC-ipsec-policy-isakmp-map1-10] quit
```

10. Apply the IPsec policy to interface GigabitEthernet 1/0/1 to protect traffic on the interface.

```
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceC-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Initiate a connection from headquarters subnet 5.5.5.0/24 to branch subnet 4.4.4.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, the traffic between the two subnets is IPsec-protected.

# Display IKE SAs on Device A.

```
[DeviceA] display ike sa
    Connection-ID   Remote                Flag         DOI
------------------------------------------------------------------
    1               2.2.2.2               RD           IPsec
Flags:
```

```
RD--READY RL--REPLACED FD-FADING RK-REKEY
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------
  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: Template
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1463
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
    sour addr: 4.4.4.0/255.255.255.0  port: 0  protocol: ip
    dest addr: 5.5.5.0/255.255.255.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 1014286405 (0x3c74c845)
      Connection ID: 1
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3590
      Max received sequence-number: 4
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 4011716027 (0xef1dedbb)
      Connection ID: 2
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3590
      Max sent sequence-number: 4
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

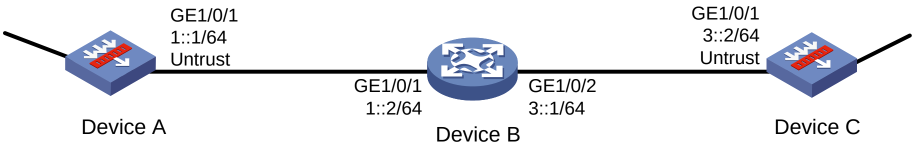
Example: Configuring IPsec for RIPng

Network configuration

As shown in Figure 22, Device A, Device B, and Device C learn IPv6 routes through RIPng.

Establish an IPsec tunnel between the devices to protect the RIPng packets transmitted in between. Specify the security protocol as ESP, the encryption algorithm as 128-bit AES, and the authentication algorithm as HMAC-SHA1 for the IPsec tunnel.

Figure 22 Network diagram



Requirements analysis

To meet the Network configuration, perform the following tasks:

1. Configure basic RIPng.

   For more information about RIPng configuration, see *Layer 3—IP Routing Configuration Guide*.
2. Configure an IPsec profile.
   - The IPsec profiles on all the devices must have IPsec transform sets that use the same security protocol, authentication and encryption algorithms, and encapsulation mode.
   - The SPI and key configured for the inbound SA and those for the outbound SA must be the same on each device.
   - The SPI and key configured for the SAs on all the devices must be the same.
3. Apply the IPsec profile to a RIPng process or to an interface.

Configuring Device A

1. Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 1::1/64
[DeviceA-GigabitEthernet1/0/1] quit
```

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 1::2.

```
[DeviceA] ipv6 route-static 3::2 64 1::2
```

3. Add interface GigabitEthernet 1/0/1 to the **Untrust** security zone.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

4. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name ipseclocalout
[DeviceA-security-policy-ipv6-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ipv6-1-ipseclocalout] destination-zone untrust
```

```
[DeviceA-security-policy-ipv6-1-ipseclocalout] source-ip-host 1::1
[DeviceA-security-policy-ipv6-1-ipseclocalout] destination-ip-host 1::2
[DeviceA-security-policy-ipv6-1-ipseclocalout] action pass
[DeviceA-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ipv6] rule name ipseclocalin
[DeviceA-security-policy-ipv6-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-ipseclocalin] source-ip-host 1::2
[DeviceA-security-policy-ipv6-2-ipseclocalin] destination-ip-host 1::1
[DeviceA-security-policy-ipv6-2-ipseclocalin] action pass
[DeviceA-security-policy-ipv6-2-ipseclocalin] quit
```

5. Configure basic RIPng.

```
[DeviceA] ripng 1
[DeviceA-ripng-1] quit
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ripng 1 enable
[DeviceA-GigabitEthernet1/0/1] quit
```

6. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

# Create and configure an IPsec transform set named **tran1**.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode transport
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

# Create and configure an IPsec profile named **profile001**.

```
[DeviceA] ipsec profile profile001 manual
[DeviceA-ipsec-profile-manual-profile001] transform-set tran1
[DeviceA-ipsec-profile-manual-profile001] sa spi outbound esp 123456
[DeviceA-ipsec-profile-manual-profile001] sa spi inbound esp 123456
[DeviceA-ipsec-profile-manual-profile001] sa string-key outbound esp simple abcdefg
[DeviceA-ipsec-profile-manual-profile001] sa string-key inbound esp simple abcdefg
[DeviceA-ipsec-profile-manual-profile001] quit
```

7. Apply the IPsec profile to RIPng process 1 to protect RIPng packets with IPsec.

```
[DeviceA] ripng 1
[DeviceA-ripng-1] enable ipsec-profile profile001
[DeviceA-ripng-1] quit
```

Configuring Device B

1. Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 1::2/64
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure basic RIPng.

```
<DeviceB> system-view
[DeviceB] ripng 1
[DeviceB-ripng-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ripng 1 enable
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ripng 1 enable
[DeviceB-GigabitEthernet1/0/2] quit
```

3. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

   # Create and configure an IPsec transform set named **tran1**.

   ```
   [DeviceB] ipsec transform-set tran1
   [DeviceB-ipsec-transform-set-tran1] encapsulation-mode transport
   [DeviceB-ipsec-transform-set-tran1] protocol esp
   [DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
   [DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
   [DeviceB-ipsec-transform-set-tran1] quit
   ```

   # Create and configure an IPsec profile named **profile001**.

   ```
   [DeviceB] ipsec profile profile001 manual
   [DeviceB-ipsec-profile-manual-profile001] transform-set tran1
   [DeviceB-ipsec-profile-manual-profile001] sa spi outbound esp 123456
   [DeviceB-ipsec-profile-manual-profile001] sa spi inbound esp 123456
   [DeviceB-ipsec-profile-manual-profile001] sa string-key outbound esp simple abcdefg
   [DeviceB-ipsec-profile-manual-profile001] sa string-key inbound esp simple abcdefg
   [DeviceB-ipsec-profile-manual-profile001] quit
   ```

4. Apply the IPsec profile to RIPng process 1 to protect RIPng packets with IPsec.

   ```
   [DeviceB] ripng 1
   [DeviceB-ripng-1] enable ipsec-profile profile001
   [DeviceB-ripng-1] quit
   ```

Configuring Device C

1. Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceC> system-view
   [DeviceC] interface gigabitethernet 1/0/1
   [DeviceC-GigabitEthernet1/0/1] ipv6 address 3::2/64
   [DeviceC-GigabitEthernet1/0/1] quit
   ```

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 3::1.

   ```
   [DeviceC] ipv6 route-static 1::1 64 3::1
   ```

3. Add interface GigabitEthernet 1/0/1 to the **Untrust** security zone.

   ```
   [DeviceC] security-zone name untrust
   [DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceC-security-zone-Untrust] quit
   ```

4. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device C to send IPsec negotiation packets to Device B.

   ```
   [DeviceC] security-policy ipv6
   ```

```
[DeviceC-security-policy-ipv6] rule name ipseclocalout
[DeviceC-security-policy-ipv6-1-ipseclocalout] source-zone local
[DeviceC-security-policy-ipv6-1-ipseclocalout] destination-zone untrust
[DeviceC-security-policy-ipv6-1-ipseclocalout] source-ip-host 3::2
[DeviceC-security-policy-ipv6-1-ipseclocalout] destination-ip-host 3::1
[DeviceC-security-policy-ipv6-1-ipseclocalout] action pass
[DeviceC-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device C to receive the IPsec negotiation packets sent from Device B.

```
[DeviceC-security-policy-ipv6] rule name ipseclocalin
[DeviceC-security-policy-ipv6-2-ipseclocalin] source-zone untrust
[DeviceC-security-policy-ipv6-2-ipseclocalin] destination-zone local
[DeviceC-security-policy-ipv6-2-ipseclocalin] source-ip-host 3::1
[DeviceC-security-policy-ipv6-2-ipseclocalin] destination-ip-host 3::2
[DeviceC-security-policy-ipv6-2-ipseclocalin] action pass
[DeviceC-security-policy-ipv6-2-ipseclocalin] quit
```

5. Configure basic RIPng.

```
[DeviceC] ripng 1
[DeviceC-ripng-1] quit
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ripng 1 enable
[DeviceC-GigabitEthernet1/0/1] quit
```

6. Configure an IPsec profile, so as to establish the IPsec tunnel to protect data:

# Create and configure an IPsec transform set named **tran1**.

```
[DeviceC] ipsec transform-set tran1
[DeviceC-ipsec-transform-set-tran1] encapsulation-mode transport
[DeviceC-ipsec-transform-set-tran1] protocol esp
[DeviceC-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceC-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceC-ipsec-transform-set-tran1] quit
```

# Create and configure an IPsec profile named **profile001**.

```
[DeviceC] ipsec profile profile001 manual
[DeviceC-ipsec-profile-manual-profile001] transform-set tran1
[DeviceC-ipsec-profile-manual-profile001] sa spi outbound esp 123456
[DeviceC-ipsec-profile-manual-profile001] sa spi inbound esp 123456
[DeviceC-ipsec-profile-manual-profile001] sa string-key outbound esp simple abcdefg
[DeviceC-ipsec-profile-manual-profile001] sa string-key inbound esp simple abcdefg
[DeviceC-ipsec-profile-manual-profile001] quit
```

7. Apply the IPsec profile to RIPng process 1 to protect RIPng packets with IPsec.

```
[DeviceC] ripng 1
[DeviceC-ripng-1] enable ipsec-profile profile001
[DeviceC-ripng-1] quit
```

Verifying the configuration

After the configuration is completed, Device A, Device B, and Device C learn IPv6 routing information through RIPng. IPsec SAs are set up successfully on the devices to protect RIPng packets. This example uses Device A to verify the configuration.

# Display the RIPng configuration.

```
[DeviceA] display ripng 1
```

```
    RIPng process : 1
       Preference : 100
       Checkzero : Enabled
       Default Cost : 0
       Maximum number of load balanced routes : 8
       Update time   :   30 secs  Timeout time         :  180 secs
       Suppress time :  120 secs  Garbage-Collect time :  120 secs
       Update output delay:   20(ms)  Output count:    3
       Graceful-restart interval:   60 secs
       Triggered Interval : 5 50 200
       Number of periodic updates sent : 186
       Number of triggered updates sent : 1
       IPsec profile name: profile001
```

The output shows that the IPsec profile **profile001** has been applied to RIPng process 1.

# Display the established IPsec SAs.

```
[DeviceA] display ipsec sa
-------------------------------
Global IPsec SA
-------------------------------

  -----------------------------
  IPsec profile: profile001
  Mode: Manual
  -----------------------------
    Encapsulation mode: transport
    [Inbound ESP SA]
      SPI: 123456 (0x3039)
      Connection ID: 1
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      No duration limit for this SA
    [Outbound ESP SA]
      SPI: 123456 (0x3039)
      Connection ID: 2
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      No duration limit for this SA
```

Example: Configuring IPsec RRI

Network configuration

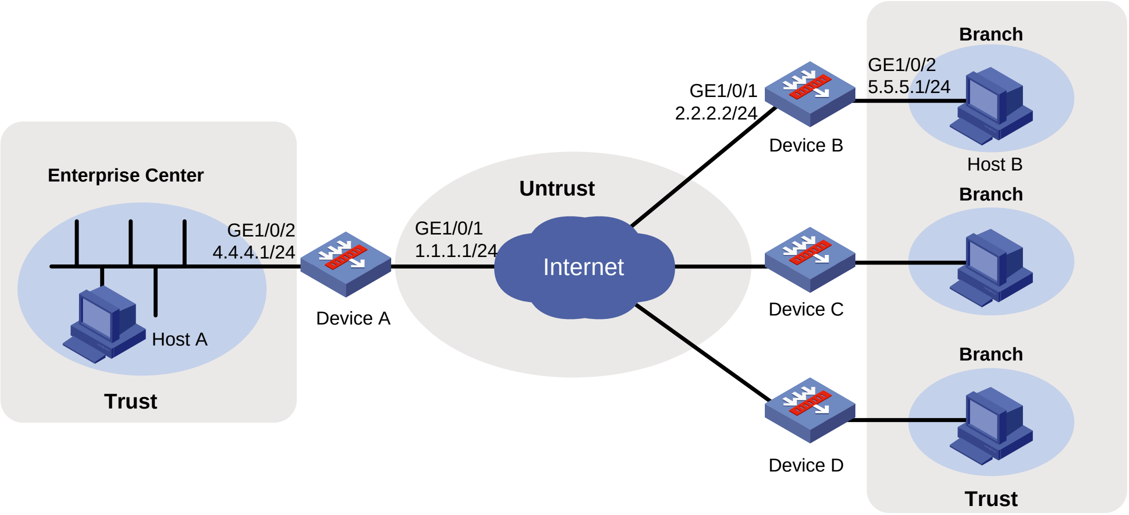
As shown in Figure 23, branches access the corporate headquarters through an IPsec VPN.

Configure the IPsec VPN as follows:

- Configure an IPsec tunnel between Device A and each branch gateway (Device B, Device C, and Device D) to protect traffic between subnets 4.4.4.0/24 and 5.5.5.0/24.
- Configure the tunnels to use the security protocol ESP, the encryption algorithm DES, and the authentication algorithm SHA1-HMAC-96. Use IKE for IPsec SA negotiation.
- Configure IKE proposal to use the preshared key authentication method, encryption algorithm 3DES, and authentication algorithm HMAC-SHA1.

- Configure IPsec RRI on Device A to automatically create static routes to the branches based on the established IPsec SAs.

Figure 23 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route, and the next hop in the route is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 24 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule name ipseclocalout
   [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
   [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
   [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
   [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
   ```

```
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 5.5.5.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 5.5.5.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 4.4.4.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

6. Configure an IKE keychain to specify the key information used for IKE communication with peer 2.2.2.2.

```
[DeviceA] ike keychain key1
[DeviceA-ike-keychain-key1] pre-shared-key address 2.2.2.2 key simple 123
[DeviceA-ike-keychain-key1] quit
```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] keychain key1
[DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.255.0
[DeviceA-ike-profile-profile1] quit
```

8. Create an IPsec policy template named **temp1**. Specify IPsec transform set **tran1** and IKE profile **profile1** for the IPsec policy template.

```
[DeviceA] ipsec policy-template temp1 1
[DeviceA-ipsec-policy-template-temp1-1] transform-set tran1
[DeviceA-ipsec-policy-template-temp1-1] ike-profile profile1
```

9. Configure IPsec RRI:

   # Enable IPsec RRI, and set the preference to 100 and the tag to 1000 for the static routes created by IPsec RRI.

```
[DeviceA-ipsec-policy-template-temp1-1] reverse-route dynamic
[DeviceA-ipsec-policy-template-temp1-1] reverse-route preference 100
[DeviceA-ipsec-policy-template-temp1-1] reverse-route tag 1000
[DeviceA-ipsec-policy-template-temp1-1] quit
```

10. Use IPsec policy template **temp1** to create an IKE-based IPsec policy to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ipsec policy map1 10 isakmp template temp1
```

11. Configure an IKE proposal to specify the security parameters used for IKE negotiation.

    # Create an IKE proposal named **1**, and specify **3DES** as the encryption algorithm, **HMAC-SHA1** as the authentication algorithm, and **pre-share** as the authentication method.

```
[DeviceA] ike proposal 1
[DeviceA-ike-proposal-1] encryption-algorithm 3des-cbc
[DeviceA-ike-proposal-1] authentication-algorithm sha
[DeviceA-ike-proposal-1] authentication-method pre-share
[DeviceA-ike-proposal-1] quit
```

12. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.3.

```
[DeviceB] ip route-static 4.4.4.0 24 2.2.2.3
[DeviceB] ip route-static 1.1.1.1 24 2.2.2.3
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 5.5.5.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 4.4.4.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 4.4.4.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 5.5.5.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an ACL to define the data flows to be protected by IPsec. IPsec RRI generates routes based on this ACL, too.

# Configure ACL 3000 to protect data from subnet 5.5.5.0/24 to subnet 4.4.4.0/24.

```
[DeviceB] acl advanced 3000
[DeviceB-acl-ipv4-adv-3000] rule permit ip source 5.5.5.0 0.0.0.255 destination 4.4.4.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3000] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of a communication must be the same.

   ```
   [DeviceB] ike keychain key1
   [DeviceB-ike-keychain-key1] pre-shared-key address 1.1.1.1 key simple 123
   [DeviceB-ike-keychain-key1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceB] ike profile profile1
   [DeviceB-ike-profile-profile1] keychain key1
   [DeviceB-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.255.0
   [DeviceB-ike-profile-profile1] quit
   ```

9. Configure an IKE-based IPsec policy to establish an IPsec tunnel to protect the specified data.

   # Create an IKE-based IPsec policy entry named **map1** and configure the following settings for the policy entry:

   - Set the sequence number to 10.
   - Specify transform set **tran1** and ACL 3000.
   - Specify the remote IP address for the tunnel as 1.1.1.1.
   - Specify IKE profile **profile1**.

   ```
   [DeviceB] ipsec policy map1 10 isakmp
   [DeviceB-ipsec-policy-isakmp-map1-10] transform-set tran1
   [DeviceB-ipsec-policy-isakmp-map1-10] security acl 3000
   [DeviceB-ipsec-policy-isakmp-map1-10] remote-address 1.1.1.1
   [DeviceB-ipsec-policy-isakmp-map1-10] ike-profile profile1
   [DeviceB-ipsec-policy-isakmp-map1-10] quit
   ```

10. Configure an IKE proposal to specify the security parameters for IKE negotiation.

    # Create an IKE proposal named **1**, and specify **3DES** as the encryption algorithm, **HMAC-SHA1** as the authentication algorithm, and **pre-share** as the authentication method.

    ```
    [DeviceB] ike proposal 1
    [DeviceB-ike-proposal-1] encryption-algorithm 3des-cbc
    [DeviceB-ike-proposal-1] authentication-algorithm sha
    [DeviceB-ike-proposal-1] authentication-method pre-share
    [DeviceB-ike-proposal-1] quit
    ```

11. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/1
    [DeviceB-GigabitEthernet1/0/1] ipsec apply policy map1
    [DeviceB-GigabitEthernet1/0/1] quit
    ```

    Make sure Device B has a route to the peer private network, with the outgoing interface as GigabitEthernet 1/0/1.

Configuring Device C and Device D

Configure Device C and Device D in the same way Device B is configured.

Verifying the configuration

1. Verify that IPsec RRI can automatically create a static route from Device A to Device B:

   # Initiate a connection from subnet 5.5.5.0/24 to subnet 4.4.4.0/24. IKE negotiation is triggered to establish IPsec SAs between Device A and Device B. (Details not shown.)

   # Verify that IPsec SAs are established on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: Template
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1463
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
    sour addr: 4.4.4.0/255.255.255.0  port: 0  protocol: ip
    dest addr: 5.5.5.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 1014286405 (0x3c74c845)
      Connection ID: 1
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3590
      Max received sequence-number: 4
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 4011716027 (0xef1dedbb)
      Connection ID: 2
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3590
```

```
Max sent sequence-number: 4
UDP encapsulation used for NAT traversal: N
Status: Active
```

# Verify that IPsec RRI has created a static route to reach Device B.

```
[DeviceA] display ip routing-table verbose
```

2. Verify that Device A can automatically create static routes to Device C and Device D in the same way that you verify the IPsec RRI feature by using Device A and Device B. (Details not shown.)

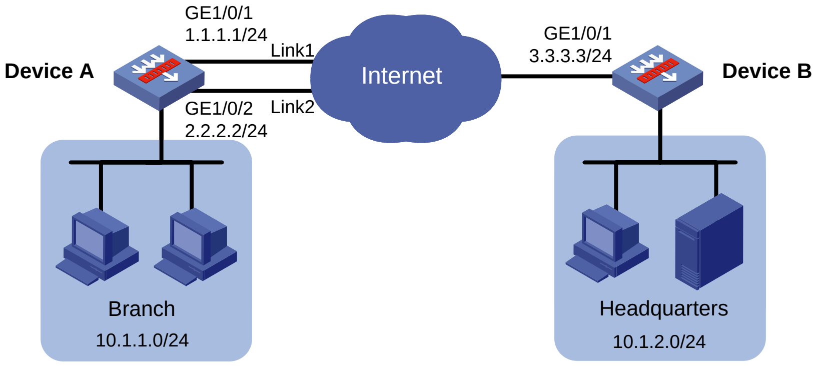
Example: Configuring IPsec smart link selection

Network configuration

As shown in Figure 24, Device A acts the IPsec gateway of the branch. Device B is the IPsec gateway of the headquarters. Configure IPsec smart link selection so the branch can establish an IPsec tunnel to the headquarters over link 1 or link 2, whichever has a better link quality.

- Device A first uses link 1 to establish the IPsec tunnel.
- When link 1 suffers high packet loss ratio or delay, Device A automatically switches traffic to the IPsec tunnel established based on link 2.

Figure 24 Network diagram



Configuring Device A

1. Configure the IP addresses and gateway addresses for GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2:

# Configure an IP address for GigabitEthernet 1/0/1 and specify the gateway address for the interface. This example uses 1.1.1.3 as the gateway address.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[DeviceA-GigabitEthernet1/0/1] gateway 1.1.1.3
[DeviceA-GigabitEthernet1/0/1] quit
```

# Configure an IP address for GigabitEthernet 1/0/2 and specify the gateway address for the interface. This example uses 2.2.2.3 as the gateway address.

```
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 2.2.2.2 24
[DeviceA-GigabitEthernet1/0/2] gateway 2.2.2.3
[DeviceA-GigabitEthernet1/0/2] quit
```

2. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
```

```
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
```

3. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.3
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.3
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

      ```
      [DeviceA-security-policy-ip] rule name untrust-trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] action pass
      [DeviceA-security-policy-ip-4-untrust-trust] quit
      [DeviceA-security-policy-ip] quit
      ```

4. Configure an IPsec smart link policy, so that the device can select a qualified link to establish an IPsec tunnel with the headquarters.

```
[DeviceA] ipsec smart-link policy policy1
[DeviceA-ipsec-smart-link-policy-policy1] link 1 interface gigabitethernet 1/0/1
remote 3.3.3.3
[DeviceA-ipsec-smart-link-policy-policy1] link 2 interface gigabitethernet 1/0/2
remote 3.3.3.3
[DeviceA-ipsec-smart-link-policy-policy1] link-switch cycles 4
[DeviceA-ipsec-smart-link-policy-policy1] smart-link enable
[DeviceA-ipsec-smart-link-policy-policy1] quit
```

5. Configure an ACL to define the data flows to be protected by IPsec.

```
[DeviceA] acl advanced 3000
[DeviceA-acl-ipv4-adv-3000] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3000] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ike keychain keychain1
[DeviceA-ike-keychain-keychain1] pre-shared-key address 3.3.3.3 24 key simple 123456
[DeviceA-ike-keychain-keychain1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] keychain keychain1
[DeviceA-ike-profile-profile1] match remote identity address 3.3.3.3 24
[DeviceA-ike-profile-profile1] quit
```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data.

```
[DeviceA] ipsec policy policy1 10 isakmp
[DeviceA-ipsec-policy-isakmp-policy1-10] security acl 3000
[DeviceA-ipsec-policy-isakmp-policy1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-policy1-10] ike-profile profile1
[DeviceA-ipsec-policy-isakmp-policy1-10] smart-link policy policy1
[DeviceA-ipsec-policy-isakmp-policy1-10] quit
```

Configuring Device B

1. Assign an IP address to interface GigabitEthernet 1/0/1, and configure the gateway address. In this example, the gateway address is the direct next hop address 3.3.3.4.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 3.3.3.3 24
[DeviceB-GigabitEthernet1/0/1] gateway 3.3.3.4
```

```
[DeviceB-GigabitEthernet1/0/1] quit
```

2. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

3. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 3.3.3.3
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 3.3.3.3
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host A and Host B:

   # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
```

```
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

4. Configure an ACL to define the data flows to be protected by IPsec.

```
[DeviceB] acl advanced 3000
[DeviceB-acl-ipv4-adv-3000] rule permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3000] rule permit ip source 3.3.3.0 0.0.0.255 destination 1.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3000] rule permit ip source 3.3.3.0 0.0.0.255 destination 2.2.2.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3000] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

6. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceB] ike keychain keychain1
[DeviceB-ike-keychain-keychain1] pre-shared-key address 0.0.0.0 0 key simple 123456
[DeviceB-ike-keychain-keychain1] quit
```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] keychain keychain1
[DeviceB-ike-profile-profile1] match remote identity address 0.0.0.0 0
[DeviceB-ike-profile-profile1] quit
```

8. Configure an IPsec policy template, which is used to create IPsec policies.

```
[DeviceB] ipsec policy-template template1 10
[DeviceB-ipsec-policy-template-template1-10] security acl 3000
[DeviceB-ipsec-policy-template-template1-10] transform-set tran1
[DeviceB-ipsec-policy-template-template1-10] local-address 3.3.3.3
[DeviceB-ipsec-policy-template-template1-10] ike-profile profile1
[DeviceB-ipsec-policy-template-template1-10] quit
```

9. Create an IKE-based IPsec policy by using IPsec policy template **template1**, so as to establish the IPsec tunnel to proctect data.

```
[DeviceB] ipsec policy policy1 10 isakmp template template1
```

10. Apply the IPsec policy to GigabitEthernet 1/0/1 to protect traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceB-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Display IPsec smart link policy information on Device A.

```
[DeviceA] display ipsec smart-link policy
---------------------------------------------------------------------------
```

```
Policy name                  : policy1
State                        :Enabled
Probe count                  :10
Probe interval               :1 sec
Probe source IP address      :1.1.1.1
Probe destination IP address :3.3.3.3
Max link switch cycles       :4
IPsec policy name            :policy1
Interface                    :GigabitEthernet1/0/1
IPsec policy sequence number :10
Link ID   Local address   Remote address   Loss(%)   Delay(ms)   State
1         1.1.1.1         3.3.3.3          0.0         1.0       Active
2         2.2.2.2         3.3.3.3          25.0        1.0       Inactive
---------------------------------------------------------------------------
```

# Display information about the IPsec policy to which IPsec smart link policy **policy1** is applied.

```
[DeviceA] display ipsec policy
-------------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet1/0/1
-------------------------------------------

  -----------------------------
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
  Traffic Flow Confidentiality: Disabled
  Security data flow: 3000
  Selector mode: standard
  Local address: 1.1.1.1
  Remote address: 3.3.3.3
  Transform set:  tran1
  IKE profile: profile1
  IKEv2 profile:
  smart-link policy: policy1
  SA trigger mode: Auto
  SA duration(time based): 3600 seconds
  SA duration(traffic based): 1843200 kilobytes
  SA idle time: 100 seconds
```

# Display the ACL rules in the ACL used by IPsec policy **policy1**.

```
[DeviceA] display acl 3000
Advanced IPv4 ACL 3000, 3 rules,
ACL's step is 5
rule 0 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
rule 5 permit ip source 1.1.1.0 0.0.0.255 destination 3.3.3.3 0 (Dynamic) (10 times matched)
```

The command output shows that an ACL rule (rule 5) is automatically added to permit the link quality probe packets.

# Verify that IPsec SAs are established on Device A.

```
[DeviceA]display ipsec sa
```

```
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------
  -----------------------------
  IPsec policy: policy1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1428
    Tunnel:
        local  address: 1.1.1.1
        remote address: 3.3.3.3
    Flow:
        sour addr: 1.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 3.3.3.3/255.255.255.255  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2443816215 (0x91a9ad17)
      Connection ID: 38654705665
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843194/3368
      Max received sequence-number: 64
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 4220315517 (0xfb8ce77d)
      Connection ID: 38654705664
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843194/3368
      Max sent sequence-number: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

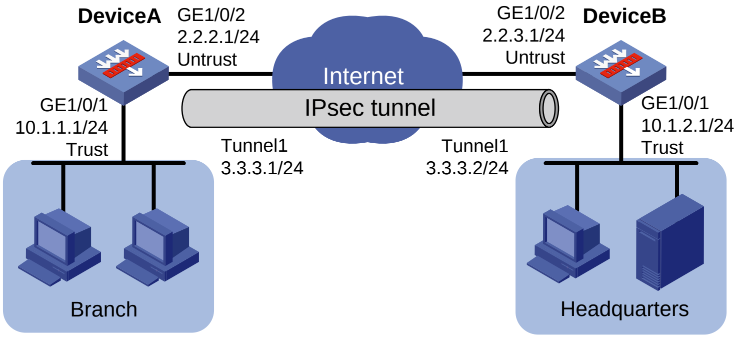
Example: Configuring IPsec tunnel interface-based IPsec for IPv4 packets (preshared key authentication)

Network configuration

As shown in Figure 25, both the branch and the headquarters use fixed IP addresses to access the Internet.

Configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between the branch (10.1.1.0/24) and the headquarters (10.1.2.0/24). This IPsec implementation ensures that the IPsec configuration of the headquarters remains stable despite of changes of the branch subnet.

Figure 25 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes.

   # Configure static routes to reach the headquarters' gateway and network. The next hop in the routes is 2.2.2.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 2.2.2.2
   [DeviceA] ip route-static 2.2.3.1 24 2.2.2.2
   ```

   # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

   ```
   [DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Untrust] import interface tunnel 1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.3.1
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.3.1
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data.

```
[DeviceA] ike keychain abc
[DeviceA-ike-keychain-abc] pre-shared-key address 2.2.3.1 255.255.255.0 key simple
123456TESTplat&!
[DeviceA-ike-keychain-abc] quit
[DeviceA] ike profile abc
[DeviceA-ike-profile-abc] keychain abc
[DeviceA-ike-profile-abc] local-identity address 2.2.2.1
[DeviceA-ike-profile-abc] match remote identity address 2.2.3.1 24
[DeviceA-ike-profile-abc] exchange-mode aggressive
```

```
[DeviceA-ike-profile-abc] quit
[DeviceA] ipsec transform-set abc
[DeviceA-ipsec-transform-set-abc] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-abc] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-abc] quit
[DeviceA] ipsec profile abc isakmp
[DeviceA-ipsec-profile-isakmp-abc] transform-set abc
[DeviceA-ipsec-profile-isakmp-abc] ike-profile abc
[DeviceA-ipsec-profile-isakmp-abc] quit
```

6. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 3.3.3.1 255.255.255.0
[DeviceA-Tunnel1] source 2.2.2.1
[DeviceA-Tunnel1] destination 2.2.3.1
[DeviceA-Tunnel1] tunnel protection ipsec profile abc
[DeviceA-Tunnel1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.2.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes

   # Configure static routes to reach the branch's gateway and network. The next hop in the routes is 2.2.3.2.

```
[DeviceB] ip route-static 10.1.1.0 24 2.2.3.2
[DeviceB] ip route-static 2.2.2.1 24 2.2.3.2
```

   # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceA] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
```

```
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data.

```
[DeviceB] ike keychain abc
[DeviceB-ike-keychain-abc] pre-shared-key address 2.2.2.1 255.255.255.0 key simple
123456TESTplat&!
[DeviceB-ike-keychain-abc] quit
[DeviceB] ike profile abc
[DeviceB-ike-profile-abc] keychain abc
[DeviceB-ike-profile-abc] local-identity address 2.2.3.1
[DeviceB-ike-profile-abc] match remote identity address 2.2.2.1 24
[DeviceB-ike-profile-abc] exchange-mode aggressive
[DeviceB-ike-profile-abc] quit
[DeviceB] ipsec transform-set abc
[DeviceB-ipsec-transform-set-abc] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-abc] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-abc] quit
```

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set abc
[DeviceB-ipsec-profile-isakmp-abc] ike-profile abc
[DeviceB-ipsec-profile-isakmp-abc] quit
```

6. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 3.3.3.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.3.1
[DeviceB-Tunnel1] destination 2.2.2.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B. After IKE negotiation succeeds, the tunnel interface will come up and traffic between the branch and the headquarters will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
<DeviceA> display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       10.1.1.1/24        --           --
GE1/0/2             up       up       2.2.2.1/24         --           --
Tun1                up       up       3.3.3.1/24         --           --
```

# Display tunnel interface information on Device A.

```
<DeviceA> display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 3.3.3.1/24 (primary)
Tunnel source 2.2.2.1, destination 2.2.3.1
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
<DeviceA> display ipsec sa
-------------------------------
Interface: Tunnel1
```

```
-------------------------------

  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 2.2.2.1
        remote address: 2.2.3.1
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in the branch subnet can ping a private IP address in the headquarters subnet successfully.

```
<DeviceA> ping -a 10.1.1.1 10.1.2.1
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
```

```
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

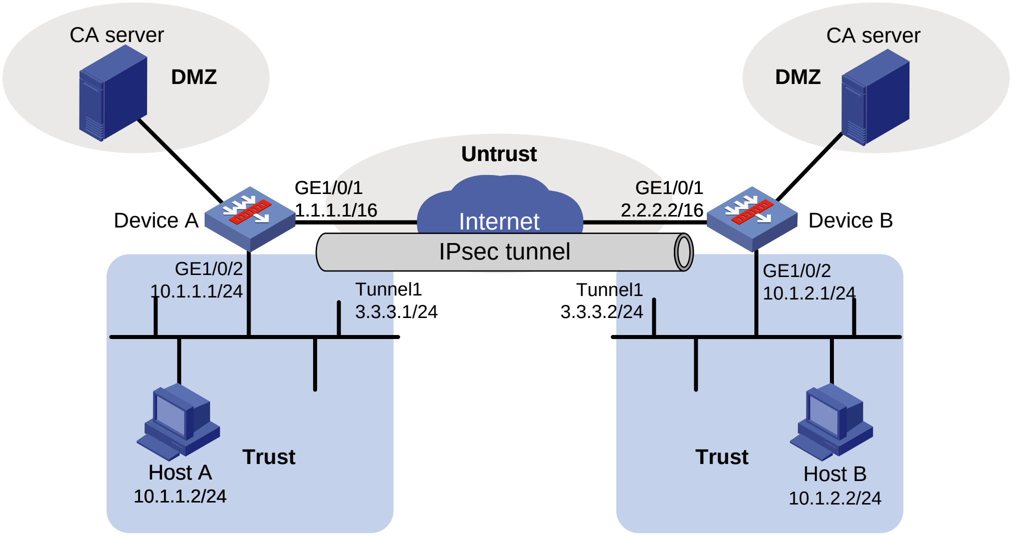
Example: Configuring IPsec tunnel interface-based IPsec for IPv4 packets (RSA signature authentication)

Network configuration

As shown in Figure 26, configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use aggressive mode for IKE negotiation phase 1 and to use RSA signature authentication. Device A acts as the initiator, and the subnet where Device A resides uses IP addresses dynamically allocated.

Figure 26 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

6. Configure a PKI entity.

```
[DeviceA] pki entity entity1
[DeviceA-pki-entity-entity1] common-name devicea
[DeviceA-pki-entity-entity1] quit
```

7. Configure a PKI domain for certificate requests.

```
[DeviceA] pki domain domain1
[DeviceA-pki-domain-domain1] public-key rsa general name rsa1
[DeviceA-pki-domain-domain1] undo crl check enable
[DeviceA-pki-domain-domain1] quit
[DeviceA] pki import domain domain1 der ca filename ca.cer
[DeviceA] pki import domain domain1 p12 local filename server.pfx
```

8. Configure a certificate-based access control policy to control user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceA] pki certificate access-control-policy policy1
[DeviceA-pki-cert-acp-policy1] rule 1 permit group1
[DeviceA] pki certificate attribute-group group1
[DeviceA-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

9. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] certificate domain domain1
[DeviceA-ike-profile-profile1] exchange-mode aggressive
[DeviceA-ike-profile-profile1] local-identity dn
[DeviceA-ike-profile-profile1] match remote certificate policy1
[DeviceA-ike-profile-profile1] quit
```

10. Configure an IKE proposal to specify the parameters used for IKE negotiation.

```
[DeviceA] ike proposal 10
[DeviceA-ike-proposal-10] authentication-algorithm md5
[DeviceA-ike-proposal-10] authentication-method rsa-signature
[DeviceA-ike-proposal-10] quit
```

11. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceA] ipsec profile abc isakmp
[DeviceA-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceA-ipsec-profile-isakmp-abc] ike-profile profile1
[DeviceA-ipsec-profile-isakmp-abc] quit
```

12. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

    # Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 3.3.3.1 255.255.255.0
[DeviceA-Tunnel1] source 1.1.1.1
[DeviceA-Tunnel1] destination 2.2.2.2
```

```
[DeviceA-Tunnel1] tunnel protection ipsec profile abc
[DeviceA-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] action pass
      ```

```
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

6. Configure a PKI entity.

```
[DeviceB] pki entity entity2
[DeviceB-pki-entity-entity2] common-name deviceb
[DeviceB-pki-entity-entity2] quit
```

7. Configure a PKI domain for certificate request.

```
[DeviceB] pki domain domain2
[DeviceB-pki-domain-domain2] public-key rsa general name rsa1
[DeviceB-pki-domain-domain2] undo crl check enable
[DeviceB-pki-domain-domain2] quit
[DeviceB] pki import domain domain2 der ca filename ca.cer
[DeviceB] pki import domain domain2 p12 local filename server.pfx
```

8. Configure a certificate-based access control policy to control user access rights.

A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceB] pki certificate access-control-policy policy1
[DeviceB-pki-cert-acp-policy1] rule 1 permit group1
[DeviceB] pki certificate attribute-group group1
[DeviceB-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

9. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile2
```

```
[DeviceB-ike-profile-profile2] certificate domain domain2
[DeviceB-ike-profile-profile2] exchange-mode aggressive
[DeviceB-ike-profile-profile2] local-identity dn
[DeviceB-ike-profile-profile2] match remote certificate policy1
[DeviceB-ike-profile-profile2] quit
```

10. Configure an IKE proposal to specify the security parameters for IKE negotiation.

```
[DeviceB] ike proposal 10
[DeviceB-ike-proposal-10] authentication-algorithm md5
[DeviceB-ike-proposal-10] authentication-method rsa-signature
[DeviceB-ike-proposal-10] quit
```

11. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceB-ipsec-profile-isakmp-abc] ike-profile profile2
[DeviceB-ipsec-profile-isakmp-abc] quit
```

12. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 3.3.3.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.2.2
[DeviceB-Tunnel1] destination 1.1.1.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B. After IKE negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       3.3.3.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
```

```
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 3.3.3.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

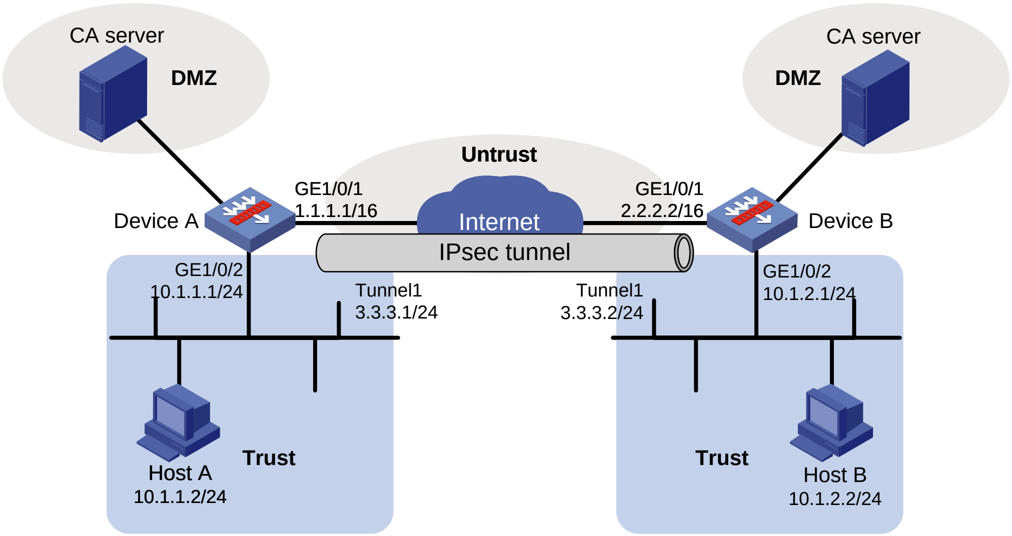
Example: Configuring IPsec tunnel interface-based IPsec for IPv4 packets (SM2-DE digital envelop authentication)

Network configuration

As shown in Figure 27, configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use GM main mode and SM2-DE digital envelop authentication for the IKE negotiation phase 1.

Figure 27 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      ```

```
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

   ```
   [DeviceA] ipsec transform-set tran1
   [DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
   [DeviceA-ipsec-transform-set-tran1] protocol esp
   [DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm sm1-cbc-128
   [DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sm3
   [DeviceA-ipsec-transform-set-tran1] quit
   ```

6. Configure a PKI entity.

   ```
   [DeviceA] pki entity entity1
   [DeviceA-pki-entity-entity1] common-name devicea
   [DeviceA-pki-entity-entity1] quit
   ```

7. Configure a PKI domain for certificate requests.

   ```
   [DeviceA] pki domain domain1
   [DeviceA-pki-domain-domain1] public-key sm2 general name sm2-1
   [DeviceA-pki-domain-domain1] undo crl check enable
   [DeviceA-pki-domain-domain1] quit
   [DeviceA] pki import domain domain1 der ca filename ca.cer
   [DeviceA] pki import domain domain1 p12 local filename server.pfx
   ```

8. Configure an IKE proposal to specify the parameters for IKE neogitiation.

   ```
   [DeviceA] ike proposal 10
   [DeviceA-ike-proposal-10] authentication-method sm2-de
   [DeviceA-ike-proposal-10] authentication-algorithm sm3
   [DeviceA-ike-proposal-10] encryption-algorithm sm1-cbc-128
   [DeviceA-ike-proposal-10] quit
   ```

9. Configure an IKE profile to specify the security parameters used to establish IKE SAs.

   ```
   [DeviceA] ike profile profile1
   [DeviceA-ike-profile-profile1] exchange-mode gm-main
   [DeviceA-ike-profile-profile1] certificate domain domain1
   [DeviceA-ike-profile-profile1] proposal 10
   [DeviceA-ike-profile-profile1] local-identity address 1.1.1.1
   [DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
   [DeviceA-ike-profile-profile1] quit
   ```

10. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

    ```
    [DeviceA] ipsec profile abc isakmp
    [DeviceA-ipsec-profile-isakmp-abc] transform-set tran1
    [DeviceA-ipsec-profile-isakmp-abc] ike-profile profile1
    ```

```
[DeviceA-ipsec-profile-isakmp-abc] quit
```

11. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

    # Create an IPsec tunnel interface, and apply the IPsec profile.

    ```
    [DeviceA] interface tunnel 1 mode ipsec
    [DeviceA-Tunnel1] ip address 3.3.3.1 255.255.255.0
    [DeviceA-Tunnel1] source 1.1.1.1
    [DeviceA-Tunnel1] destination 2.2.2.2
    [DeviceA-Tunnel1] tunnel protection ipsec profile abc
    [DeviceA-Tunnel1] quit
    ```

    # Add the IPsec tunnel interface to security zone **Untrust**.

    ```
    [DeviceA] security-zone name untrust
    [DeviceA-security-zone-Untrust] import interface tunnel 1
    [DeviceA-security-zone-Untrust] quit
    ```

    # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

    ```
    [DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
    ```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm sm1-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sm3
[DeviceB-ipsec-transform-set-tran1] quit
```

6. Configure a PKI entity.

```
[DeviceB] pki entity entity2
[DeviceB-pki-entity-entity2] common-name deviceb
[DeviceB-pki-entity-entity2] quit
```

7. Configure a PKI domain for certificate request.

```
[DeviceB] pki domain domain2
[DeviceB-pki-domain-domain2] public-key sm2 general name sm2-1
[DeviceB-pki-domain-domain2] undo crl check enable
[DeviceB-pki-domain-domain2] quit
[DeviceB] pki import domain domain2 der ca filename ca.cer
[DeviceB] pki import domain domain2 p12 local filename server.pfx
```

8. Configure an IKE proposal to specify the parameters for IKE neogitiation.

```
[DeviceB] ike proposal 10
```

```
[DeviceB-ike-proposal-10] authentication-method sm2-de
[DeviceB-ike-proposal-10] authentication-algorithm sm3
[DeviceB-ike-proposal-10] encryption-algorithm sm1-cbc-128
[DeviceB-ike-proposal-10] quit
```

9. Configure an IKE profile to specify the security parameters used to establish IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] exchange-mode gm-main
[DeviceB-ike-profile-profile1] certificate domain domain2
[DeviceB-ike-profile-profile1] proposal 10
[DeviceB-ike-profile-profile1] local-identity address 2.2.2.2
[DeviceB-ike-profile-profile1] match remote identity address 1.1.1.1 255.255.0.0
[DeviceB-ike-profile-profile1] quit
```

10. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceB-ipsec-profile-isakmp-abc] ike-profile profile1
[DeviceB-ipsec-profile-isakmp-abc] quit
```

11. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 3.3.3.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.2.2
[DeviceB-Tunnel1] destination 1.1.1.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B. After IKE negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       3.3.3.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
```

```
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 3.3.3.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
```

```
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

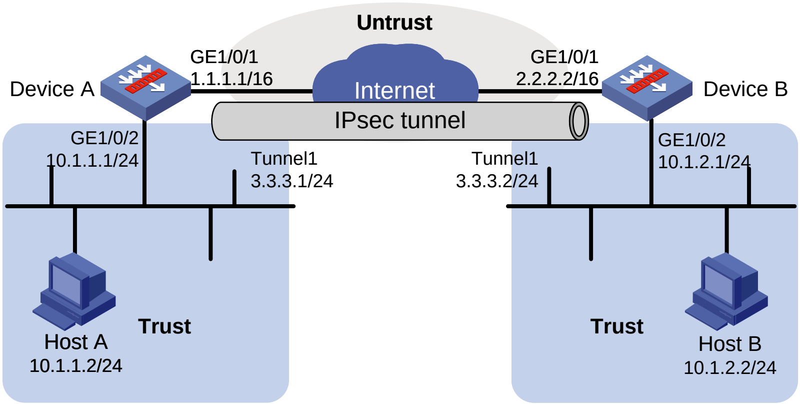
Example: Configuring IPsec tunnel interface-based IPsec for IPv4 packets (IKEv2 with preshared key authentication)

Network configuration

As shown in Figure 28, configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Specify the encapsulation mode as tunnel, the security protocol as ESP, the encryption algorithm as DES-CBC, and the authentication algorithm as HMAC-SHA1.
- Set up SAs through IKEv2 negotiation.

Figure 28 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

      ```
      [DeviceA-security-policy-ip] rule name untrust-trust
      ```

```
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

6. Configure an IKEv2 keychain to specify the key information used for IKEv2 communication.

```
[DeviceA] ikev2 keychain keychain1
[DeviceA-ikev2-keychain-keychain1] peer peer1
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 16
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
[DeviceA-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext abcde
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceA-ikev2-keychain-keychain1] quit
```

7. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] authentication-method local pre-share
[DeviceA-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceA-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
[DeviceA-ikev2-profile-profile1] quit
```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceA] ipsec profile abc isakmp
[DeviceA-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceA-ipsec-profile-isakmp-abc] ikev2-profile profile1
[DeviceA-ipsec-profile-isakmp-abc] quit
```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 3.3.3.1 255.255.255.0
[DeviceA-Tunnel1] source 1.1.1.1
[DeviceA-Tunnel1] destination 2.2.2.2
[DeviceA-Tunnel1] tunnel protection ipsec profile abc
[DeviceA-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] action pass
      [DeviceB-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
      ```

```
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

6. Configure an IKEv2 keychain to specify the key information used for IKEv2 communication.

```
[DeviceB] ikev2 keychain keychain1
[DeviceB-ikev2-keychain-keychain1] peer peer1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] address 1.1.1.1 16
[DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 1.1.1.1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext abcde
[DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceB-ikev2-keychain-keychain1] quit
```

7. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceB] ikev2 profile profile1
[DeviceB-ikev2-profile-profile1] authentication-method local pre-share
[DeviceB-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceB-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] match remote identity address 1.1.1.1 255.255.0.0
[DeviceA-ikev2-profile-profile1] quit
```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceB-ipsec-profile-isakmp-abc] ikev2-profile profile1
[DeviceB-ipsec-profile-isakmp-abc] quit
```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 3.3.3.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.2.2
[DeviceB-Tunnel1] destination 1.1.1.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKEv2 negotiation with Device B. After IKEv2 negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       3.3.3.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 3.3.3.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
```

```
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

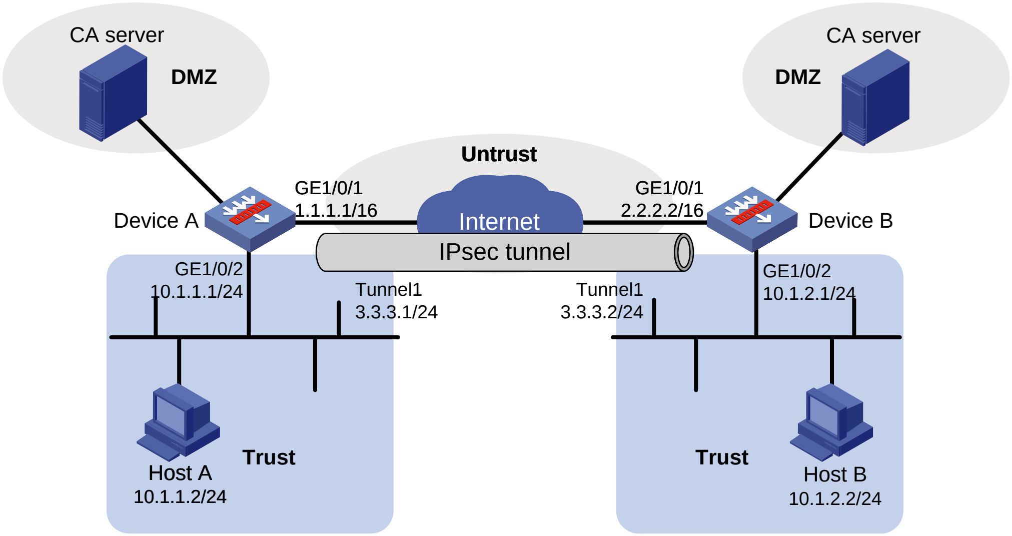
Example: Configuring IPsec tunnel interface-based IPsec for IPv4 packets (IKEv2 with RSA signature authentication)

Network configuration

As shown in Figure 29, configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to set up SAs through IKEv2 negotiation and to use RSA signature authentication. Device A acts as the initiator, and the subnet where Device A resides uses IP addresses dynamically allocated.

Figure 29 Network diagram



Prerequisites

Device A has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Device B has obtained CA certificate **ca.cer** and local certificate **server.pfx**.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   ```

```
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:
   - **a.** Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

     # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

     ```
     [DeviceA] security-policy ip
     [DeviceA-security-policy-ip] rule name ipseclocalout
     [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
     [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
     [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
     [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
     [DeviceA-security-policy-ip-1-ipseclocalout] action pass
     [DeviceA-security-policy-ip-1-ipseclocalout] quit
     ```

     # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

     ```
     [DeviceA-security-policy-ip] rule name ipseclocalin
     [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
     [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
     [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
     [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
     [DeviceA-security-policy-ip-2-ipseclocalin] action pass
     [DeviceA-security-policy-ip-2-ipseclocalin] quit
     ```

   - **b.** Configure rules to permit the traffic between Host A and Host B:

     # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

     ```
     [DeviceA-security-policy-ip] rule name trust-untrust
     [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
     [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
     [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
     [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
     [DeviceA-security-policy-ip-3-trust-untrust] action pass
     [DeviceA-security-policy-ip-3-trust-untrust] quit
     ```

     # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

     ```
     [DeviceA-security-policy-ip] rule name untrust-trust
     [DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
     [DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
     [DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
     [DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
     [DeviceA-security-policy-ip-4-untrust-trust] action pass
     [DeviceA-security-policy-ip-4-untrust-trust] quit
     [DeviceA-security-policy-ip] quit
     ```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

   ```
   [DeviceA] ipsec transform-set tran1
   [DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
   [DeviceA-ipsec-transform-set-tran1] protocol esp
   [DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
   ```

```
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

6. Configure a PKI entity.

```
[DeviceA] pki entity entity1
[DeviceA-pki-entity-entity1] common-name devicea
[DeviceA-pki-entity-entity1] quit
```

7. Configure a PKI domain for certificate requests.

```
[DeviceA] pki domain domain1
[DeviceA-pki-domain-domain1] public-key rsa general name rsa1
[DeviceA-pki-domain-domain1] undo crl check enable
[DeviceA-pki-domain-domain1] quit
[DeviceA] pki import domain domain1 der ca filename ca.cer
[DeviceA] pki import domain domain1 p12 local filename server.pfx
```

8. Configure a certificate-based access control policy to control user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceA] pki certificate access-control-policy policy1
[DeviceA-pki-cert-acp-policy1] rule 1 permit group1
[DeviceA] pki certificate attribute-group group1
[DeviceA-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

9. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] authentication-method local rsa-signature
[DeviceA-ikev2-profile-profile1] authentication-method remote rsa-signature
[DeviceA-ikev2-profile-profile1] certificate domain domain1
[DeviceA-ikev2-profile-profile1] identity local dn
[DeviceA-ikev2-profile-profile1] match remote certificate policy1
[DeviceA-ikev2-profile-profile1] quit
```

10. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceA] ipsec profile abc isakmp
[DeviceA-ipsec-profile-isakmp-abc] transform-set tran1
[DeviceA-ipsec-profile-isakmp-abc] ikev2-profile profile1
[DeviceA-ipsec-profile-isakmp-abc] quit
```

11. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

    # Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 3.3.3.1 255.255.255.0
[DeviceA-Tunnel1] source 1.1.1.1
[DeviceA-Tunnel1] destination 2.2.2.2
[DeviceA-Tunnel1] tunnel protection ipsec profile abc
[DeviceA-Tunnel1] quit
```

    # Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] quit
```

    # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 1.1.1.1 16 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] action pass
      [DeviceB-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] action pass
      ```

```
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

6. Configure a PKI entity.

```
[DeviceB] pki entity entity2
[DeviceB-pki-entity-entity2] common-name deviceb
[DeviceB-pki-entity-entity2] quit
```

7. Configure a PKI domain for certificate request.

```
[DeviceB] pki domain domain2
[DeviceB-pki-domain-domain2] public-key rsa general name rsa1
[DeviceB-pki-domain-domain2] undo crl check enable
[DeviceB-pki-domain-domain2] quit
[DeviceB] pki import domain domain2 der ca filename ca.cer
[DeviceB] pki import domain domain2 p12 local filename server.pfx
```

8. Configure a certificate-based access control policy to control user access rights.

   A peer certificate is regarded valid only if it contains the specified string (**1** in this example) in the DN attribute of the subject name field.

```
[DeviceB] pki certificate access-control-policy policy1
[DeviceB-pki-cert-acp-policy1] rule 1 permit group1
[DeviceB] pki certificate attribute-group group1
[DeviceB-pki-cert-attribute-group-group1] attribute 1 subject-name dn ctn 1
```

9. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceB] ikev2 profile profile2
[DeviceB-ikev2-profile-profile2] authentication-method local rsa-signature
[DeviceB-ikev2-profile-profile2] authentication-method remote rsa-signature
[DeviceB-ikev2-profile-profile2] certificate domain domain2
[DeviceB-ikev2-profile-profile2] identity local dn
[DeviceB-ikev2-profile-profile2] match remote certificate policy1
[DeviceB-ikev2-profile-profile2] quit
```

10. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set tran1
```

```
[DeviceB-ipsec-profile-isakmp-abc] ikev2-profile profile2
[DeviceB-ipsec-profile-isakmp-abc] quit
```

11. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

    # Create an IPsec tunnel interface, and apply the IPsec profile.

    ```
    [DeviceB] interface tunnel 1 mode ipsec
    [DeviceB-Tunnel1] ip address 3.3.3.2 255.255.255.0
    [DeviceB-Tunnel1] source 2.2.2.2
    [DeviceB-Tunnel1] destination 1.1.1.1
    [DeviceB-Tunnel1] tunnel protection ipsec profile abc
    [DeviceB-Tunnel1] quit
    ```

    # Add the IPsec tunnel interface to security zone **Untrust**.

    ```
    [DeviceB] security-zone name untrust
    [DeviceB-security-zone-Untrust] import interface tunnel 1
    [DeviceB-security-zone-Untrust] quit
    ```

    # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

    ```
    [DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
    ```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKEv2 negotiation with Device B. After IKEv2 negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       3.3.3.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 3.3.3.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
```

```
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
```

```
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

Example: Configuring an IKE-based IPsec NAT traversal tunnel between gateways

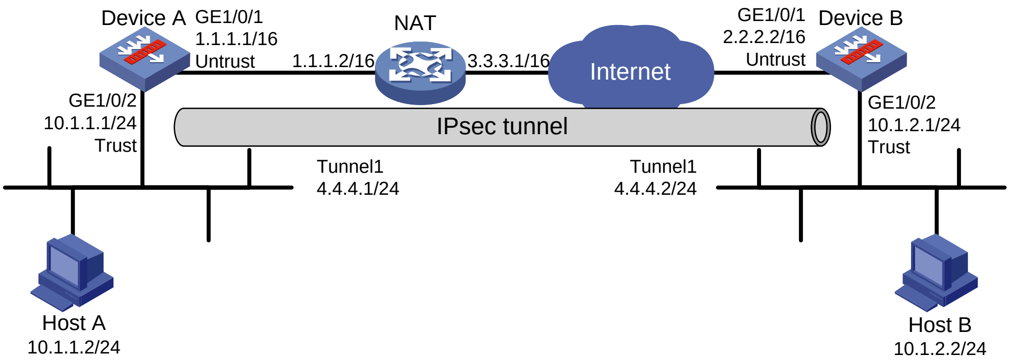
Network configuration

Device A is behind the NAT device. Hosts behind Device A use public IP address 3.3.3.1 to access the external network.

Configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKE proposal for the aggressive IKE negotiation to set up the IPsec SAs.
- Configure the two devices to use the preshared key authentication method for the IKE negotiation phase 1.

Figure 30 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   ```

```
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin
      [DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-2-ipseclocalin] action pass
      [DeviceA-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit the traffic between Host A and Host B:

      # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

      ```
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

      ```
      [DeviceA-security-policy-ip] rule name untrust-trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-4-untrust-trust] action pass
      [DeviceA-security-policy-ip-4-untrust-trust] quit
      [DeviceA-security-policy-ip] quit
      ```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   ```
   [DeviceA] ipsec transform-set transform1
   [DeviceA-ipsec-transform-set-transform1] protocol esp
   [DeviceA-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
   [DeviceA-ipsec-transform-set-transform1] esp authentication-algorithm md5
   ```

```
[DeviceA-ipsec-transform-set-transform1] quit
```

6. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceA] ike keychain keychain1
   [DeviceA-ike-keychain-keychain1] pre-shared-key address 2.2.2.2 255.255.0.0 key
   simple 12345zxcvb!@#$%ZXCVB
   [DeviceA-ike-keychain-keychain1] quit
   ```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceA] ike profile profile1
   [DeviceA-ike-profile-profile1] keychain keychain1
   [DeviceA-ike-profile-profile1] exchange-mode aggressive
   [DeviceA-ike-profile-profile1] local-identity fqdn www.devicea.com
   [DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
   [DeviceA-ike-profile-profile1] quit
   ```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

   ```
   [DeviceA] ipsec profile abc isakmp
   [DeviceA-ipsec-profile-isakmp-abc] transform-set transform1
   [DeviceA-ipsec-profile-isakmp-abc] ike-profile profile1
   [DeviceA-ipsec-profile-isakmp-abc] quit
   ```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

   # Create an IPsec tunnel interface, and apply the IPsec profile.

   ```
   [DeviceA] interface tunnel 1 mode ipsec
   [DeviceA-Tunnel1] ip address 4.4.4.1 255.255.255.0
   [DeviceA-Tunnel1] source 1.1.1.1
   [DeviceA-Tunnel1] destination 2.2.2.2
   [DeviceA-Tunnel1] tunnel protection ipsec profile abc
   [DeviceA-Tunnel1] quit
   ```

   # Add the IPsec tunnel interface to security zone **Untrust**.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface tunnel 1
   [DeviceA-security-zone-Untrust] quit
   ```

   # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

   ```
   [DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
   ```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 3.3.3.1 16 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   ```

```
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.1
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.1
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] action pass
      [DeviceB-security-policy-ip-2-ipseclocalin] quit
      ```

   b. Configure rules to permit traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] action pass
      [DeviceB-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

      ```
      [DeviceB-security-policy-ip] rule name untrust-trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-4-untrust-trust] action pass
      [DeviceB-security-policy-ip-4-untrust-trust] quit
      [DeviceB-security-policy-ip] quit
      ```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   ```
   [DeviceB] ipsec transform-set transform1
   [DeviceB-ipsec-transform-set-transform1] protocol esp
   ```

```
[DeviceB-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceB-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceB-ipsec-transform-set-transform1] quit
```

6. Configure an IKE keychain to specify the key information used for IKE communication.

   This example specifies **12345zxcvb!@#$%ZXCVB** in plain text as the preshared key to be used with the remote peer at 1.1.1.1. The source address of packets from 1.1.1.1 is translated into 3.3.3.1 by the NAT device, so IP address of the remote peer is specified as 3.3.3.1.

```
[DeviceB]ike keychain keychain1
[DeviceB-ike-keychain-keychain1] pre-shared-key address 3.3.3.1 255.255.0.0 key
simple 12345zxcvb!@#$%ZXCVB
[DeviceB-ike-keychain-keychain1] quit
```

7. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] keychain keychain1
[DeviceB-ike-profile-profile1] exchange-mode aggressive
[DeviceB-ike-profile-profile1] match remote identity fqdn www.devicea.com
[DeviceB-ike-profile-profile1] quit
```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set transform1
[DeviceB-ipsec-profile-isakmp-abc] ike-profile profile1
[DeviceB-ipsec-profile-isakmp-abc] quit
```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

   # Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 4.4.4.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.2.2
[DeviceB-Tunnel1] destination 3.3.3.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

   # Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

   # Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B. After IKE negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       4.4.4.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1444
Internet address: 4.4.4.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

Example: Configuring headquarters to use dual links to establish IPsec tunnels with branches

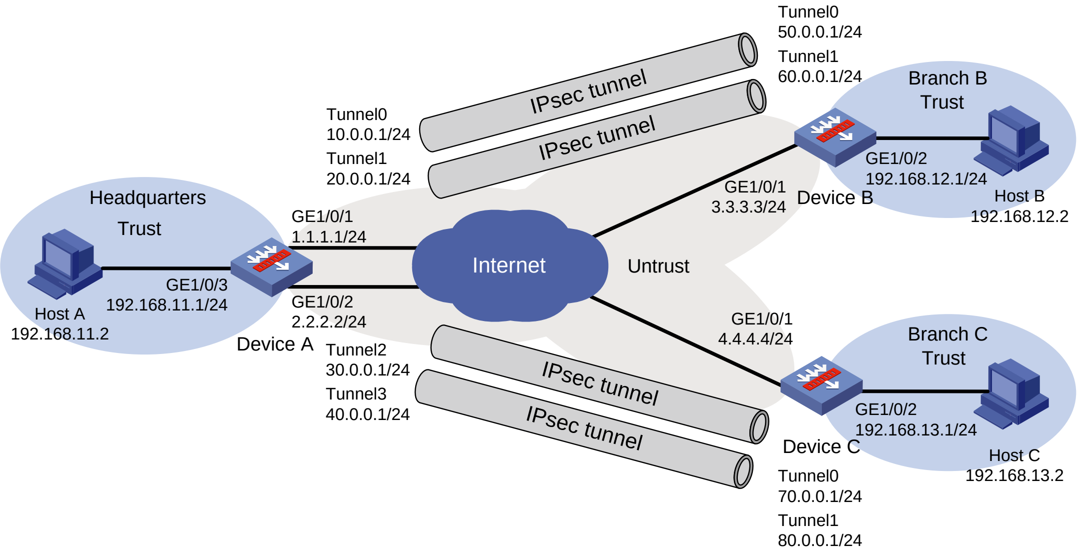
Network configuration

As shown in Figure 31, the headquarters gateway device Device A has two links connected to the Internet. Branch gateway devices Device B and Device C each has one link connected to the Internet.

Configure IPsec tunnel interface-based IPsec on the devices protect the traffic between the headquarters and the branches.

The headquarters and the branches select high-quality, low-latency links based on the NQA test results to dynamically establish IPsec tunnels.

Figure 31 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Create IPsec tunnel interfaces:

# Create IPsec tunnel interface Tunnel 0.

```
[DeviceA] interface tunnel 0 mode ipsec
[DeviceA-Tunnel0] ip address 10.0.0.1 255.255.255.0
[DeviceA-Tunnel0] source 1.1.1.1
[DeviceA-Tunnel0] destination 3.3.3.3
[DeviceA-Tunnel0] quit
```

# Create IPsec tunnel interface Tunnel 1.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 20.0.0.1 255.255.255.0
[DeviceA-Tunnel1] source 2.2.2.2
[DeviceA-Tunnel1] destination 3.3.3.3
[DeviceA-Tunnel1] quit
```

# Create IPsec tunnel interface Tunnel 2.

```
[DeviceA] interface tunnel 2 mode ipsec
[DeviceA-Tunnel2] ip address 30.0.0.1 255.255.255.0
[DeviceA-Tunnel2] source 1.1.1.1
[DeviceA-Tunnel2] destination 4.4.4.4
[DeviceA-Tunnel2] quit
```

# Create IPsec tunnel interface Tunnel 3.

```
[DeviceA] interface tunnel 3 mode ipsec
[DeviceA-Tunnel3] ip address 40.0.0.1 255.255.255.0
```

```
[DeviceA-Tunnel3] source 2.2.2.2
[DeviceA-Tunnel3] destination 4.4.4.4
[DeviceA-Tunnel3] quit
```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] import interface tunnel 0
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] import interface tunnel 2
[DeviceA-security-zone-Untrust] import interface tunnel 3
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:
   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B and Device C.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.3
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 4.4.4.4
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B and Device C.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.3
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 4.4.4.4
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

   b. Configure rules to permit the traffic between Host A and Host B and between Host A and Host C:

   # Configure a rule named **trust-untrust** to permit the packets from Host A to Host B and Host C.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 192.168.11.0 24
```

```
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 192.168.12.0
24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 192.168.13.0
24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B and Host C to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 192.168.12.0 24
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 192.168.13.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 192.168.11.0
24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure IPsec profiles to establish IPsec tunnels to protect the specified data:

# Configure IPsec profile **t0**.

```
[DeviceA] ike keychain t0
[DeviceA-ike-keychain-t0] pre-shared-key address 3.3.3.3 255.255.255.0 key simple
123456TESTplat&!
[DeviceA-ike-keychain-t0] quit
[DeviceA] ike profile t0
[DeviceA-ike-profile-t0] keychain t0
[DeviceA-ike-profile-t0] match local address Tunnel0
[DeviceA-ike-profile-t0] match remote identity address 3.3.3.3 24
[DeviceA-ike-profile-t0] exchange-mode aggressive
[DeviceA-ike-profile-t0] dpd interval 30 periodic
[DeviceA-ike-profile-t0] quit
[DeviceA] ipsec transform-set t0
[DeviceA-ipsec-transform-set-t0] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-t0] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-t0] quit
[DeviceA] ipsec profile t0 isakmp
[DeviceA-ipsec-profile-isakmp-t0] transform-set t0
[DeviceA-ipsec-profile-isakmp-t0] ike-profile t0
[DeviceA-ipsec-profile-isakmp-t0] quit
```

# Configure IPsec profile **t1**.

```
[DeviceA] ike keychain t1
[DeviceA-ike-keychain-t1] pre-shared-key address 3.3.3.3 255.255.255.0 key simple
123456TESTplat&!
[DeviceA-ike-keychain-t1] quit
[DeviceA] ike profile t1
[DeviceA-ike-profile-t1] keychain t1
[DeviceA-ike-profile-t1] match local address Tunnel1
[DeviceA-ike-profile-t1] match remote identity address 3.3.3.3 24
[DeviceA-ike-profile-t1] exchange-mode aggressive
```

```
[DeviceA-ike-profile-t1] dpd interval 30 periodic
[DeviceA-ike-profile-t1] quit
[DeviceA] ipsec transform-set t1
[DeviceA-ipsec-transform-set-t1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-t1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-t1] quit
[DeviceA] ipsec profile t1 isakmp
[DeviceA-ipsec-profile-isakmp-t1] transform-set t1
[DeviceA-ipsec-profile-isakmp-t1] ike-profile t1
[DeviceA-ipsec-profile-isakmp-t1] quit
```

# Configure IPsec profile **t2**.

```
[DeviceA] ike keychain t2
[DeviceA-ike-keychain-t2] pre-shared-key address 4.4.4.4 255.255.255.0 key simple
123456TESTplat&!
[DeviceA-ike-keychain-t2] quit
[DeviceA] ike profile t2
[DeviceA-ike-profile-t2] keychain t2
[DeviceA-ike-profile-t2] match local address Tunnel2
[DeviceA-ike-profile-t2] match remote identity address 4.4.4.4 24
[DeviceA-ike-profile-t2] exchange-mode aggressive
[DeviceA-ike-profile-t2] dpd interval 30 periodic
[DeviceA-ike-profile-t2] quit
[DeviceA] ipsec transform-set t2
[DeviceA-ipsec-transform-set-t2] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-t2] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-t2] quit
[DeviceA] ipsec profile t2 isakmp
[DeviceA-ipsec-profile-isakmp-t2] transform-set t2
[DeviceA-ipsec-profile-isakmp-t2] ike-profile t2
[DeviceA-ipsec-profile-isakmp-t2] quit
```

# Configure IPsec profile **t3**.

```
[DeviceA] ike keychain t3
[DeviceA-ike-keychain-t3] pre-shared-key address 4.4.4.4 255.255.255.0 key simple
123456TESTplat&!
[DeviceA-ike-keychain-t3] quit
[DeviceA] ike profile t3
[DeviceA-ike-profile-t3] keychain t3
[DeviceA-ike-profile-t3] match local address Tunnel3
[DeviceA-ike-profile-t3] match remote identity address 4.4.4.4 24
[DeviceA-ike-profile-t3] exchange-mode aggressive
[DeviceA-ike-profile-t3] dpd interval 30 periodic
[DeviceA-ike-profile-t3] quit
[DeviceA] ipsec transform-set t3
[DeviceA-ipsec-transform-set-t3] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-t3] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-t3] quit
[DeviceA] ipsec profile t3 isakmp
[DeviceA-ipsec-profile-isakmp-t3] transform-set t3
[DeviceA-ipsec-profile-isakmp-t3] ike-profile t3
```

```
[DeviceA-ipsec-profile-isakmp-t3] quit
```

6. Configure IPsec tunnel interfaces, which encapsulate the traffic to be protected with IPsec.

# Configure IPsec tunnel interface Tunnel 0 to use IPsec profile **t0** to establish an IPsec tunnel.

```
[DeviceA] interface tunnel 0
[DeviceA-Tunnel0] tunnel protection ipsec profile t0
[DeviceA-Tunnel0] quit
```

# Configure IPsec tunnel interface Tunnel 1 to use IPsec profile **t1** to establish an IPsec tunnel.

```
[DeviceA] interface tunnel 1
[DeviceA-Tunnel1] tunnel protection ipsec profile t1
[DeviceA-Tunnel1] quit
```

# Configure IPsec tunnel interface Tunnel 2 to use IPsec profile **t2** to establish an IPsec tunnel.

```
[DeviceA] interface tunnel 2
[DeviceA-Tunnel2] tunnel protection ipsec profile t2
[DeviceA-Tunnel2] quit
```

# Configure IPsec tunnel interface Tunnel 3 to use IPsec profile **t3** to establish an IPsec tunnel.

```
[DeviceA] interface tunnel 3
[DeviceA-Tunnel3] tunnel protection ipsec profile t3
[DeviceA-Tunnel3] quit
```

7. Configure NQA operation and Track collaboration to probe the status of links:

# Configure NQA operation with administrator name **admin** and operation tag **test1**.

```
[DeviceA] nqa entry admin test1
[DeviceA-nqa-admin-test1] type icmp-echo
[DeviceA-nqa-admin-test1-icmp-echo] destination ip 3.3.3.3
[DeviceA-nqa-admin-test1-icmp-echo] frequency 3000
[DeviceA-nqa-admin-test1-icmp-echo] history-record enable
[DeviceA-nqa-admin-test1-icmp-echo] next-hop ip 1.1.1.2
[DeviceA-nqa-admin-test1-icmp-echo] probe count 5
[DeviceA-nqa-admin-test1-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceA-nqa-admin-test1-icmp-echo] quit
[DeviceA] nqa schedule admin test1 start-time now lifetime forever
```

# Create track entry 110 and associate it with reaction entry 1 of NQA operation **admin-test1**.

```
[DeviceA] track 110 nqa entry admin test1 reaction 1
[DeviceA-track-110] quit
```

# Configure NQA operation with administrator name **admin** and operation tag **test2**.

```
[DeviceA] nqa entry admin test2
[DeviceA-nqa-admin-test2] type icmp-echo
[DeviceA-nqa-admin-test2-icmp-echo] destination ip 3.3.3.3
[DeviceA-nqa-admin-test2-icmp-echo] frequency 3000
[DeviceA-nqa-admin-test2-icmp-echo] history-record enable
[DeviceA-nqa-admin-test2-icmp-echo] probe count 5
[DeviceA-nqa-admin-test2-icmp-echo] next-hop ip 2.2.2.3
[DeviceA-nqa-admin-test2-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceA-nqa-admin-test2-icmp-echo] quit
[DeviceA] nqa schedule admin test2 start-time now lifetime forever
```

# Create track entry 120 and associate it with reaction entry 1 of NQA operation **admin-test2**.

```
[DeviceA] track 120 nqa entry admin test2 reaction 1
```

```
[DeviceA-track-120] quit
```

# Configure NQA operation with administrator name **admin** and operation tag **test3**.

```
[DeviceA] nqa entry admin test3
[DeviceA-nqa-admin-test3] type icmp-echo
[DeviceA-nqa-admin-test3-icmp-echo] destination ip 4.4.4.4
[DeviceA-nqa-admin-test3-icmp-echo] frequency 3000
[DeviceA-nqa-admin-test3-icmp-echo] history-record enable
[DeviceA-nqa-admin-test3-icmp-echo] probe count 5
[DeviceA-nqa-admin-test3-icmp-echo] next-hop ip 1.1.1.2
[DeviceA-nqa-admin-test3-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceA-nqa-admin-test3-icmp-echo] quit
[DeviceA] nqa schedule admin test3 start-time now lifetime forever
```

# Create track entry 130 and associate it with reaction entry 1 of NQA operation **admin-test3**.

```
[DeviceA] track 130 nqa entry admin test3 reaction 1
[DeviceA-track-130] quit
```

# Configure NQA operation with administrator name **admin** and operation tag **test4**.

```
[DeviceA] nqa entry admin test4
[DeviceA-nqa-admin-test4] type icmp-echo
[DeviceA-nqa-admin-test4-icmp-echo] destination ip 4.4.4.4
[DeviceA-nqa-admin-test4-icmp-echo] frequency 3000
[DeviceA-nqa-admin-test4-icmp-echo] history-record enable
[DeviceA-nqa-admin-test4-icmp-echo] probe count 5
[DeviceA-nqa-admin-test4-icmp-echo] next-hop ip 2.2.2.3
[DeviceA-nqa-admin-test4-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceA-nqa-admin-test4-icmp-echo] quit
[DeviceA] nqa schedule admin test4 start-time now lifetime forever
```

# Create track entry 140 and associate it with reaction entry 1 of NQA operation **admin-test4**.

```
[DeviceA] track 140 nqa entry admin test4 reaction 1
[DeviceA-track-140] quit
```

8. Configure settings for routing.

   This example configures static routes.

   # Configure static routes to direct the traffic to be protected to the IPsec tunnel interfaces. Configure static routes to reach the branches' gateways and the next hops in the routes are 1.1.1.2 and 2.2.2.3.

```
[DeviceA] ip route-static 192.168.12.0 24 tunnel 0 track 110 preference 100
[DeviceA] ip route-static 192.168.12.0 24 tunnel 1 track 120 preference 110
[DeviceA] ip route-static 192.168.13.0 24 tunnel 2 track 130 preference 100
[DeviceA] ip route-static 192.168.13.0 24 tunnel 3 track 140 preference 110
[DeviceA] ip route-static 3.3.3.3 24 1.1.1.2 track 110 preference 100
[DeviceA] ip route-static 3.3.3.3 24 2.2.2.3 track 120 preference 110
[DeviceA] ip route-static 4.4.4.4 24 1.1.1.2 track 130 preference 100
[DeviceA] ip route-static 4.4.4.4 24 2.2.2.3 track 140 preference 110
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
```

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 3.3.3.3 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Create IPsec tunnel interfaces:

   # Create IPsec tunnel interface Tunnel 0.

   ```
   [DeviceB] interface tunnel 0 mode ipsec
   [DeviceB-Tunnel0] ip address 50.0.0.1 255.255.255.0
   [DeviceB-Tunnel0] source 3.3.3.3
   [DeviceB-Tunnel0] destination 1.1.1.1
   [DeviceB-Tunnel0] quit
   ```

   # Create IPsec tunnel interface Tunnel 1.

   ```
   [DeviceB] interface tunnel 1 mode ipsec
   [DeviceB-Tunnel1] ip address 60.0.0.1 255.255.255.0
   [DeviceB-Tunnel1] source 3.3.3.3
   [DeviceB-Tunnel1] destination 2.2.2.2
   [DeviceB-Tunnel1] quit
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] import interface tunnel 0
   [DeviceB-security-zone-Untrust] import interface tunnel 1
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout
      [DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 3.3.3.3
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout] quit
      ```

      # Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin
      [DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
      [DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 3.3.3.3
      ```

```
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 192.168.12.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 192.168.11.0
24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 192.168.11.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 192.168.12.0
24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure IPsec profiles to establish IPsec tunnels to protect the specified data:

# Configure IPsec profile **t0**.

```
[DeviceB] ike keychain t0
[DeviceB-ike-keychain-t0] pre-shared-key address 1.1.1.1 255.255.255.0 key simple
123456TESTplat&!
[DeviceB-ike-keychain-t0] quit
[DeviceB] ike profile t0
[DeviceB-ike-profile-t0] keychain t0
[DeviceB-ike-profile-t0] match local address Tunnel0
[DeviceB-ike-profile-t0] match remote identity address 1.1.1.1 24
[DeviceB-ike-profile-t0] exchange-mode aggressive
[DeviceB-ike-profile-t0] dpd interval 30 periodic
[DeviceB-ike-profile-t0] quit
[DeviceB] ipsec transform-set t0
[DeviceB-ipsec-transform-set-t0] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-t0] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-t0] quit
[DeviceB] ipsec profile t0 isakmp
[DeviceB-ipsec-profile-isakmp-t0] transform-set t0
[DeviceB-ipsec-profile-isakmp-t0] ike-profile t0
[DeviceB-ipsec-profile-isakmp-t0] quit
```

# Configure IPsec profile **t1**.

```
[DeviceB] ike keychain t1
[DeviceB-ike-keychain-t1] pre-shared-key address 2.2.2.2 255.255.255.0 key simple
123456TESTplat&!
[DeviceB-ike-keychain-t1] quit
```

```
[DeviceB] ike profile t1
[DeviceB-ike-profile-t1] keychain t1
[DeviceB-ike-profile-t1] match local address Tunnel1
[DeviceB-ike-profile-t1] match remote identity address 2.2.2.2 24
[DeviceB-ike-profile-t1] exchange-mode aggressive
[DeviceB-ike-profile-t1] dpd interval 30 periodic
[DeviceB-ike-profile-t1] quit
[DeviceB] ipsec transform-set t1
[DeviceB-ipsec-transform-set-t1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-t1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-t1] quit
[DeviceB] ipsec profile t1 isakmp
[DeviceB-ipsec-profile-isakmp-t1] transform-set t1
[DeviceB-ipsec-profile-isakmp-t1] ike-profile t1
[DeviceB-ipsec-profile-isakmp-t1] quit
```

6. Configure IPsec tunnel interfaces, which encapsulate the traffic to be protected with IPsec.

   # Configure IPsec tunnel interface Tunnel 0 to use IPsec profile **t0** to establish an IPsec tunnel.

```
[DeviceB] interface tunnel 0
[DeviceB-Tunnel0] tunnel protection ipsec profile t0
[DeviceB-Tunnel0] quit
```

   # Configure IPsec tunnel interface Tunnel 1 to use IPsec profile **t1** to establish an IPsec tunnel.

```
[DeviceB] interface tunnel 1
[DeviceB-Tunnel1] tunnel protection ipsec profile t1
[DeviceB-Tunnel1] quit
```

7. Configure NQA operation and Track collaboration to probe the status of links:

   # Configure NQA operation with administrator name **admin** and operation tag **test1**.

```
[DeviceB] nqa entry admin test1
[DeviceB-nqa-admin-test1] type icmp-echo
[DeviceB-nqa-admin-test1-icmp-echo] destination ip 1.1.1.1
[DeviceB-nqa-admin-test1-icmp-echo] frequency 3000
[DeviceB-nqa-admin-test1-icmp-echo] history-record enable
[DeviceB-nqa-admin-test1-icmp-echo] probe count 5
[DeviceB-nqa-admin-test1-icmp-echo] next-hop ip 3.3.3.4
[DeviceB-nqa-admin-test1-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceB-nqa-admin-test1-icmp-echo] quit
[DeviceB] nqa schedule admin test1 start-time now lifetime forever
```

   # Create track entry 110 and associate it with reaction entry 1 of NQA operation **admin-test1**.

```
[DeviceB] track 110 nqa entry admin test1 reaction 1
[DeviceB-track-110] quit
```

   # Configure NQA operation with administrator name **admin** and operation tag **test2**.

```
[DeviceB] nqa entry admin test2
[DeviceB-nqa-admin-test2] type icmp-echo
[DeviceB-nqa-admin-test2-icmp-echo] destination ip 2.2.2.2
[DeviceB-nqa-admin-test2-icmp-echo] frequency 3000
[DeviceB-nqa-admin-test2-icmp-echo] history-record enable
[DeviceB-nqa-admin-test2-icmp-echo] probe count 5
[DeviceB-nqa-admin-test2-icmp-echo] next-hop ip 3.3.3.4
```

```
[DeviceB-nqa-admin-test2-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceB-nqa-admin-test2-icmp-echo] quit
[DeviceB] nqa schedule admin test2 start-time now lifetime forever
```

# Create track entry 120 and associate it with reaction entry 1 of NQA operation **admin-test2**.

```
[DeviceB] track 120 nqa entry admin test2 reaction 1
[DeviceB-track-120] quit
```

8. Configure settings for routing.

   This example configures static routes.

   # Configure static routes to direct the traffic to be protected to the IPsec tunnel interfaces. Configure static routes to reach the headquarters' gateway and the next hop in the routes is 3.3.3.4.

```
[DeviceB] ip route-static 192.168.11.0 24 tunnel 0 track 110 preference 100
[DeviceB] ip route-static 192.168.11.0 24 tunnel 1 track 120 preference 110
[DeviceB] ip route-static 1.1.1.1 24 3.3.3.4
[DeviceB] ip route-static 2.2.2.2 24 3.3.3.4
```

Configuring Device C

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ip address 4.4.4.4 255.255.255.0
[DeviceC-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Create IPsec tunnel interfaces:

   # Create IPsec tunnel interface Tunnel 0.

```
[DeviceC] interface tunnel 0 mode ipsec
[DeviceC-Tunnel0] ip address 70.0.0.1 255.255.255.0
[DeviceC-Tunnel0] source 4.4.4.4
[DeviceC-Tunnel0] destination 1.1.1.1
[DeviceC-Tunnel0] quit
```

   # Create IPsec tunnel interface Tunnel 1.

```
[DeviceC] interface tunnel 1 mode ipsec
[DeviceC-Tunnel1] ip address 80.0.0.1 255.255.255.0
[DeviceC-Tunnel1] source 4.4.4.4
[DeviceC-Tunnel1] destination 2.2.2.2
[DeviceC-Tunnel1] quit
```

3. Add interfaces to security zones.

```
[DeviceC] security-zone name trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
[DeviceC] security-zone name untrust
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceC-security-zone-Untrust] import interface tunnel 0
[DeviceC-security-zone-Untrust] import interface tunnel 1
[DeviceC-security-zone-Untrust] quit
```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device C to send IPsec negotiation packets to Device A.

```
[DeviceC] security-policy ip
[DeviceC-security-policy-ip] rule name ipseclocalout
[DeviceC-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceC-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceC-security-policy-ip-1-ipseclocalout] source-ip-host 4.4.4.4
[DeviceC-security-policy-ip-1-ipseclocalout] destination-ip-host 1.1.1.1
[DeviceC-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
[DeviceC-security-policy-ip-1-ipseclocalout] action pass
[DeviceC-security-policy-ip-1-ipseclocalout] quit
```

   # Configure a rule named **ipseclocalin** to allow Device C to receive the IPsec negotiation packets sent from Device A.

```
[DeviceC-security-policy-ip] rule name ipseclocalin
[DeviceC-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceC-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceC-security-policy-ip-2-ipseclocalin] source-ip-host 1.1.1.1
[DeviceC-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceC-security-policy-ip-2-ipseclocalin] destination-ip-host 4.4.4.4
[DeviceC-security-policy-ip-2-ipseclocalin] action pass
[DeviceC-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host C and Host A:

   # Configure a rule named **trust-untrust** to permit the packets from Host C to Host A.

```
[DeviceC-security-policy-ip] rule name trust-untrust
[DeviceC-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceC-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceC-security-policy-ip-3-trust-untrust] source-ip-subnet 192.168.13.0 24
[DeviceC-security-policy-ip-3-trust-untrust] destination-ip-subnet 192.168.11.0
24
[DeviceC-security-policy-ip-3-trust-untrust] action pass
[DeviceC-security-policy-ip-3-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit the packets from Host A to Host C.

```
[DeviceC-security-policy-ip] rule name untrust-trust
[DeviceC-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceC-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceC-security-policy-ip-4-untrust-trust] source-ip-subnet 192.168.11.0 24
[DeviceC-security-policy-ip-4-untrust-trust] destination-ip-subnet 192.168.13.0
24
[DeviceC-security-policy-ip-4-untrust-trust] action pass
[DeviceC-security-policy-ip-4-untrust-trust] quit
[DeviceC-security-policy-ip] quit
```

5. Configure IPsec profiles to establish IPsec tunnels to protect the specified data:

   # Configure IPsec profile **t0**.

```
[DeviceC] ike keychain t0
[DeviceC-ike-keychain-t0] pre-shared-key address 1.1.1.1 255.255.255.0 key simple
123456TESTplat&!
[DeviceC-ike-keychain-t0] quit
```

```
[DeviceC] ike profile t0
[DeviceC-ike-profile-t0] keychain t0
[DeviceC-ike-profile-t0] match local address Tunnel0
[DeviceC-ike-profile-t0] match remote identity address 1.1.1.1 24
[DeviceC-ike-profile-t0] exchange-mode aggressive
[DeviceC-ike-profile-t0] dpd interval 30 periodic
[DeviceC-ike-profile-t0] quit
[DeviceC] ipsec transform-set t0
[DeviceC-ipsec-transform-set-t0] esp encryption-algorithm aes-cbc-128
[DeviceC-ipsec-transform-set-t0] esp authentication-algorithm sha1
[DeviceC-ipsec-transform-set-t0] quit
[DeviceC] ipsec profile t0 isakmp
[DeviceC-ipsec-profile-isakmp-t0] transform-set t0
[DeviceC-ipsec-profile-isakmp-t0] ike-profile t0
[DeviceC-ipsec-profile-isakmp-t0] quit
```

# Configure IPsec profile **t1**.

```
[DeviceC] ike keychain t1
[DeviceC-ike-keychain-t1] pre-shared-key address 2.2.2.2 255.255.255.0 key simple
123456TESTplat&!
[DeviceC-ike-keychain-t1] quit
[DeviceC] ike profile t1
[DeviceC-ike-profile-t1] keychain t1
[DeviceC-ike-profile-t1] match local address Tunnel1
[DeviceC-ike-profile-t1] match remote identity address 2.2.2.2 24
[DeviceC-ike-profile-t1] exchange-mode aggressive
[DeviceC-ike-profile-t1] dpd interval 30 periodic
[DeviceC-ike-profile-t1] quit
[DeviceC] ipsec transform-set t1
[DeviceC-ipsec-transform-set-t1] esp encryption-algorithm aes-cbc-128
[DeviceC-ipsec-transform-set-t1] esp authentication-algorithm sha1
[DeviceC-ipsec-transform-set-t1] quit
[DeviceC] ipsec profile t1 isakmp
[DeviceC-ipsec-profile-isakmp-t1] transform-set t1
[DeviceC-ipsec-profile-isakmp-t1] ike-profile t1
[DeviceC-ipsec-profile-isakmp-t1] quit
```

6. Configure IPsec tunnel interfaces, which encapsulate the traffic to be protected with IPsec.

# Configure IPsec tunnel interface Tunnel 0 to use IPsec profile **t0** to establish an IPsec tunnel.

```
[DeviceC] interface tunnel 0
[DeviceC-Tunnel0] tunnel protection ipsec profile t0
[DeviceC-Tunnel0] quit
```

# Configure IPsec tunnel interface Tunnel 1 to use IPsec profile **t1** to establish an IPsec tunnel.

```
[DeviceC] interface tunnel 1
[DeviceC-Tunnel1] tunnel protection ipsec profile t1
[DeviceC-Tunnel1] quit
```

7. Configure NQA operation and Track collaboration to probe the status of links:

# Configure NQA operation with administrator name **admin** and operation tag **test1**.

```
[DeviceC] nqa entry admin test1
[DeviceC-nqa-admin-test1] type icmp-echo
```

```
[DeviceC-nqa-admin-test1-icmp-echo] destination ip 1.1.1.1
[DeviceC-nqa-admin-test1-icmp-echo] frequency 3000
[DeviceC-nqa-admin-test1-icmp-echo] history-record enable
[DeviceC-nqa-admin-test1-icmp-echo] probe count 5
[DeviceC-nqa-admin-test1-icmp-echo] next-hop ip 4.4.4.5
[DeviceC-nqa-admin-test1-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceC-nqa-admin-test1-icmp-echo] quit
[DeviceC] nqa schedule admin test1 start-time now lifetime forever
```

# Create track entry 110 and associate it with reaction entry 1 of NQA operation **admin-test1**.

```
[DeviceC] track 110 nqa entry admin test1 reaction 1
[DeviceC-track-110] quit
```

# Configure NQA operation with administrator name **admin** and operation tag **test2**.

```
[DeviceC] nqa entry admin test2
[DeviceC-nqa-admin-test2] type icmp-echo
[DeviceC-nqa-admin-test2-icmp-echo] destination ip 2.2.2.2
[DeviceC-nqa-admin-test2-icmp-echo] frequency 3000
[DeviceC-nqa-admin-test2-icmp-echo] history-record enable
[DeviceC-nqa-admin-test2-icmp-echo] probe count 5
[DeviceC-nqa-admin-test2-icmp-echo] next-hop ip 4.4.4.5
[DeviceC-nqa-admin-test2-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 3 action-type trigger-only
[DeviceC-nqa-admin-test2-icmp-echo] quit
[DeviceC] nqa schedule admin test2 start-time now lifetime forever
```

# Create track entry 120 and associate it with reaction entry 1 of NQA operation **admin-test2**.

```
[DeviceC] track 120 nqa entry admin test2 reaction 1
[DeviceC-track-120] quit
```

8. Configure settings for routing.

   This example configures static routes.

   # Configure static routes to direct the traffic to be protected to the IPsec tunnel interfaces. Configure static routes to reach the headquarters' gateway and the next hop in the routes is 4.4.4.5.

```
[DeviceC] ip route-static 192.168.11.0 24 tunnel 0 track 110 preference 100
[DeviceC] ip route-static 192.168.11.0 24 tunnel 1 track 120 preference 110
[DeviceC] ip route-static 1.1.1.1 24 4.4.4.5
[DeviceC] ip route-static 2.2.2.2 24 4.4.4.5
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B and Device C. After IKE negotiation succeeds, the tunnel interfaces on the devices will come up and traffic between the branch and the headquarters will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
<DeviceA> display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/24          --           --
GE1/0/2             up       up       2.2.2.2/24          --           --
```

```
GE1/0/3             up       up       192.168.11.1/24     --           --
Tun0                up       up       10.0.0.1/24         --           --
Tun1                up       up       20.0.0.1/24         --           --
Tun2                up       up       30.0.0.1/24         --           --
Tun3                up       up       40.0.0.1/24         --           --
```

# Display routing information o Device A.

```
<DeviceA> display ip routing-table

Destinations : 14        Routes : 14

Destination/Mask   Proto   Pre Cost        NextHop         Interface
1.1.1.0/24         Direct  0   0           1.1.1.1         GE1/0/1
1.1.1.255/32       Direct  0   0           1.1.1.1         GE1/0/1
2.2.2.0/24         Direct  0   0           2.2.2.2         GE1/0/2
2.2.2.255/32       Direct  0   0           2.2.2.2         GE1/0/2
3.3.3.0/24         Static  100 0           1.1.1.2         GE1/0/1
4.4.4.0/24         Static  100 0           1.1.1.2         GE1/0/1
10.0.0.0/24        Direct  0   0           10.0.0.1        Tun0
10.0.0.255/32      Direct  0   0           10.0.0.1        Tun0
20.0.0.0/24        Direct  0   0           20.0.0.1        Tun1
20.0.0.255/32      Direct  0   0           20.0.0.1        Tun1
192.168.11.0/24    Direct  0   0           192.168.11.1    GE1/0/3
192.168.11.255/32  Direct  0   0           192.168.11.1    GE1/0/3
192.168.12.0/24    Static  100 0           0.0.0.0         Tun0
192.168.13.0/24    Static  100 0           0.0.0.0         Tun2
```

# Verify that Host A can ping Host B. IPsec traffic is transmitted over the link of Tunnel 0.

```
C:\Users\hosta> ping 192.168.12.2
(Details not shown.)
```

# After the link connected to GigabitEthernet 1/0/1 of Device A fails, Host A still can ping Host B. The IPsec traffic transmitted over the link of Tunnel 1.

```
C:\Users\hosta> ping 192.168.12.2
(Details not shown.)
```

# Display interface status information on Device A.

```
<DeviceA> display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             down     down     1.1.1.1/24          --           --
GE1/0/2             up       up       2.2.2.2/24          --           --
GE1/0/3             up       up       192.168.11.1/24     --           --
Tun0                down     down     10.0.0.1/24         --           --
Tun1                up       up       20.0.0.1/24         --           --
Tun2                down     down     30.0.0.1/24         --           --
Tun3                up       up       40.0.0.1/24         --           --
```

# Display routing information on Device A.

```
<DeviceA> display ip routing-table
```

```
Destinations : 10        Routes : 10

Destination/Mask   Proto   Pre Cost        NextHop         Interface
2.2.2.0/24         Direct  0   0           2.2.2.2         GE1/0/2
2.2.2.255/32       Direct  0   0           2.2.2.2         GE1/0/2
3.3.3.0/24         Static  110 0           2.2.2.3         GE1/0/2
4.4.4.0/24         Static  110 0           2.2.2.3         GE1/0/2
20.0.0.0/24        Direct  0   0           20.0.0.1        Tun1
20.0.0.255/32      Direct  0   0           20.0.0.1        Tun1
192.168.11.0/24    Direct  0   0           192.168.11.1    GE1/0/3
192.168.11.255/32  Direct  0   0           192.168.11.1    GE1/0/3
192.168.12.0/24    Static  110 0           0.0.0.0         Tun1
192.168.13.0/24    Static  110 0           0.0.0.0         Tun3
```

# After the link connected to GigabitEthernet 1/0/1 of Device A recovers, IPsec traffic is switched back to the link of Tunnel 0.

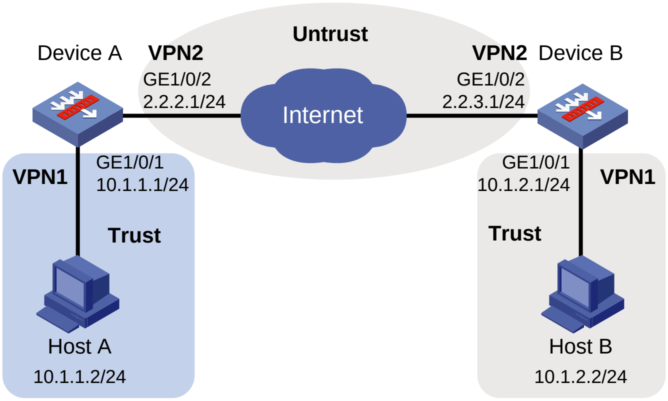
Example: Configuring IPsec with multiple VPN instances

Network configuration

As shown in Figure 32, establish an IPsec tunnel between Device A and Device B to protect data flows between subnet 10.1.1.0/24 and subnet 10.1.2.0/24. Configure the tunnel as follows:

- Specify the encapsulation mode as tunnel, the security protocol as ESP, the encryption algorithm as 128-bit AES, and the authentication algorithm as HMAC-SHA1.
- Set up SAs through IKE negotiation.
- The intranet and exanet of Device A and Device B belong to different VPN instances.

Figure 32 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] ip vpn-instance vpn1
   [DeviceA-vpn-instance-vpn1] route-distinguisher 100:1
   [DeviceA-vpn-instance-vpn1] quit
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip binding vpn-instance vpn1
   ```

```
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] ip vpn-instance vpn2
[DeviceA-vpn-instance-vpn2] route-distinguisher 100:2
[DeviceA-vpn-instance-vpn2] quit
[DeviceA] interface
[DeviceA-] ip binding vpn-instance vpn2
[DeviceA-] ip address 2.2.2.1 255.255.255.0
[DeviceA-] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[DeviceA] ip route-static vpn-instance vpn1 10.1.2.0 24 vpn-instance vpn2 2.2.2.2
[DeviceA] ip route-static vpn-instance vpn2 2.2.3.1 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
```

4. Configure a security policy:

a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ipseclocalout
[DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.1
[DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.3.1
[DeviceA-security-policy-ip-1-ipseclocalout] action pass
[DeviceA-security-policy-ip-1-ipseclocalout] vrf vpn2
[DeviceA-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.3.1
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalout] vrf vpn2
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
```

```
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-ipseclocalout] vrf vpn1
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-ipseclocalout] vrf vpn1
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

   # Configure an IPv4 advanced ACL to identify data flows from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip vpn-instance vpn1 source 10.1.1.0
0.0.0.255 destination 10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ike keychain keychain1 vpn-instance vpn2
[DeviceA-ike-keychain-keychain1] pre-shared-key address 2.2.3.1 255.255.255.0 key
simple 123456TESTplat&!
[DeviceA-ike-keychain-keychain1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceA] ike profile profile1
[DeviceA-ike-profile-profile1] keychain keychain1
[DeviceA-ike-profile-profile1] match remote identity address 2.2.3.1 255.255.255.0
vpn-instance vpn2
[DeviceA-ike-profile-profile1] inside-vpn vpn-instance vpn1
[DeviceA-ike-profile-profile1] quit
```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

# Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

```
[DeviceA] ipsec policy map1 10 isakmp
[DeviceA-ipsec-policy-isakmp-map1-10] security acl 3101
[DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
[DeviceA-ipsec-policy-isakmp-map1-10] local-address 2.2.2.1
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.3.1
[DeviceA-ipsec-policy-isakmp-map1-10] ike-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

```
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/2] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] ip vpn-instance vpn1
[DeviceB-vpn-instance-vpn1] route-distinguisher 200:1
[DeviceB-vpn-instance-vpn1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip binding vpn-instance vpn1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.2.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] ip vpn-instance vpn2
[DeviceB-vpn-instance-vpn2] route-distinguisher 200:2
[DeviceB-vpn-instance-vpn2] quit
[DeviceB] interface
[DeviceB-] ip binding vpn-instance vpn2
[DeviceB-] ip address 2.2.3.1 255.255.255.0
[DeviceB-] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.3.2.

```
[DeviceB] ip route-static vpn-instance vpn1 10.1.1.0 24 vpn-instance vpn2 2.2.3.2
[DeviceB] ip route-static vpn-instance vpn2 2.2.2.1 24 2.2.3.2
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

# Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] vrf vpn2
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalout] vrf vpn2
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-ipseclocalout] vrf vpn1
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-ipseclocalout] vrf vpn1
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure an IPv4 advanced ACL to identify data flows from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip vpn-instance vpn1 source 10.1.2.0
0.0.0.255 destination 10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

   The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

```
[DeviceB] ike keychain keychain1 vpn-instance vpn2
[DeviceB-ike-keychain-keychain1] pre-shared-key address 2.2.2.1 255.255.255.0 key
simple 123456TESTplat&!
[DeviceB-ike-keychain-keychain1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] keychain keychain1
[DeviceB-ike-profile-profile1] match remote identity address 2.2.2.1 255.255.255.0
vpn-instance vpn2
[DeviceB-ike-profile-profile1] inside-vpn vpn-instance vpn1
[DeviceB-ike-profile-profile1] quit
```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

```
[DeviceB] ipsec policy use1 10 isakmp
[DeviceB-ipsec-policy-isakmp-use1-10] security acl 3101
[DeviceB-ipsec-policy-isakmp-use1-10] transform-set tran1
[DeviceB-ipsec-policy-isakmp-use1-10] local-address 2.2.3.1
[DeviceB-ipsec-policy-isakmp-use1-10] remote-address 2.2.2.1
[DeviceB-ipsec-policy-isakmp-use1-10] ike-profile profile1
[DeviceB-ipsec-policy-isakmp-use1-10] quit
```

10. Apply the IPsec policy to GigabitEthernet 1/0/2 to protect traffic on the interface.

```
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IPsec SAs are successfully negotiated by IKE, the traffic between the two subnets is IPsec-protected.

# Display IPsec SAs on Device A and Device B. This example uses Device A to verify the configuration.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet 1/0/2
-------------------------------


  -----------------------------
```

```
IPsec policy: map1
Sequence number: 10
Alisa: map1-10
Mode: ISAKMP
-----------------------------
  Tunnel id: 0
  Encapsulation mode: tunnel
  Perfect Forward Secrecy:
  Inside VPN: vpn1
  Extended Sequence Numbers enable: N
  Traffic Flow Confidentiality enable: N
  Transmitting entity: Initiator
  Path MTU: 1443
  Tunnel:
      local  address: 2.2.3.1
      remote address: 2.2.2.1
  Flow:
      sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
      dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip
  [Inbound ESP SAs]
    SPI: 3769702703 (0xe0b1192f)
    Connection ID: 90194313219
    Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
    SA duration (kilobytes/sec): 1843200/3600
    SA remaining duration (kilobytes/sec): 2300/797
    Max received sequence-number: 1
    Anti-replay check enable: Y
    Anti-replay window size:
    UDP encapsulation used for NAT traversal: N
    Status: Active
  [Outbound ESP SAs]
    SPI: 3840956402 (0xe4f057f2)
    Connection ID: 64424509441
    Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
    SA duration (kilobytes/sec): 1843200/3600
    SA remaining duration (kilobytes/sec): 2312/797
    Max sent sequence-number: 1
    UDP encapsulation used for NAT traversal: N
    Status: Active
```

Example: Configuring an IKEv2-based IPsec NAT traversal tunnel between gateways

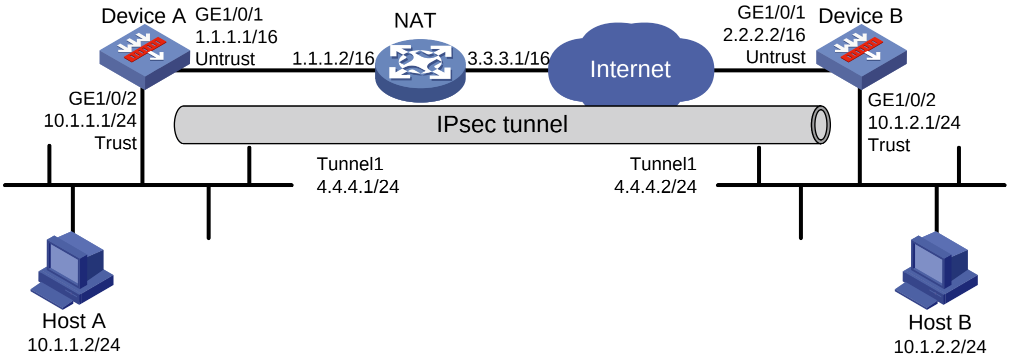
Network configuration

Device A is behind the NAT device. Hosts behind Device A use public IP address 3.3.3.1 to access the external network.

Configure IPsec tunnel interface-based IPsec on Device A and Device B to protect the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKEv2 proposal and the default IKEv2 policy in IKEv2 negotiation to set up IPsec SAs.
- Configure the two devices to use the preshared key authentication method in IKEv2 negotiation phase 1.

Figure 33 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.0.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 1.1.1.2.

   ```
   [DeviceA] ip route-static 2.2.2.2 16 1.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout
      [DeviceA-security-policy-ip-1-ipseclocalout] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout] source-ip-host 1.1.1.1
      [DeviceA-security-policy-ip-1-ipseclocalout] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout] quit
      ```

# Configure a rule named **ipseclocalin** to allow Device A to receive the IPsec negotiation packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name ipseclocalin
[DeviceA-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceA-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceA-security-policy-ip-2-ipseclocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-ipseclocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-ipseclocalin] action pass
[DeviceA-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceA] ipsec transform-set transform1
[DeviceA-ipsec-transform-set-transform1] protocol esp
[DeviceA-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceA-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceA-ipsec-transform-set-transform1] quit
```

6. Configure an IKEv2 keychain to specify the key information used for IKEv2 communication.

The preshared key used by both sides of the communication must be the same.

```
[DeviceA] ikev2 keychain keychain1
[DeviceA-ikev2-keychain-keychain1] peer peer1
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 16
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
[DeviceA-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext 123
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceA-ikev2-keychain-keychain1] quit
```

7. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceA] ikev2 profile profile1
[DeviceA-ikev2-profile-profile1] keychain keychain1
[DeviceA-ikev2-profile-profile1] identity local fqdn www.devicea.com
```

```
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.0.0
[DeviceA-ikev2-profile-profile1] authentication-method local pre-share
[DeviceA-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceA-ikev2-profile-profile1] quit
```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceA] ipsec profile abc isakmp
[DeviceA-ipsec-profile-isakmp-abc] transform-set transform1
[DeviceA-ipsec-profile-isakmp-abc] ikev2-profile profile1
[DeviceA-ipsec-profile-isakmp-abc] quit
```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceA] interface tunnel 1 mode ipsec
[DeviceA-Tunnel1] ip address 4.4.4.1 255.255.255.0
[DeviceA-Tunnel1] source 1.1.1.1
[DeviceA-Tunnel1] destination 2.2.2.2
[DeviceA-Tunnel1] tunnel protection ipsec profile abc
[DeviceA-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 tunnel 1
```

Configuring Device B

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.2.2.2 255.255.0.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

This example configures static routes, and the next hop in the routes is 2.2.2.1.

```
[DeviceB] ip route-static 3.3.3.1 16 2.2.2.1
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

   # Configure a rule named **ipseclocalout** to allow Device B to send IPsec negotiation packets to Device A.

```
[DeviceB] security-policy ip
```

```
[DeviceB-security-policy-ip] rule name ipseclocalout
[DeviceB-security-policy-ip-1-ipseclocalout] source-zone local
[DeviceB-security-policy-ip-1-ipseclocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ipseclocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-ipseclocalout] destination-ip-host 3.3.3.1
[DeviceB-security-policy-ip-1-ipseclocalout] action pass
[DeviceB-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Device B to receive the IPsec negotiation packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name ipseclocalin
[DeviceB-security-policy-ip-2-ipseclocalin] source-zone untrust
[DeviceB-security-policy-ip-2-ipseclocalin] destination-zone local
[DeviceB-security-policy-ip-2-ipseclocalin] source-ip-host 3.3.3.1
[DeviceB-security-policy-ip-2-ipseclocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-ipseclocalin] action pass
[DeviceB-security-policy-ip-2-ipseclocalin] quit
```

b. Configure rules to permit traffic between Host B and Host A:

# Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-3-trust-untrust] action pass
[DeviceB-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

5. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings at both sides of the IPsec tunnel must be the same.

```
[DeviceB] ipsec transform-set transform1
[DeviceB-ipsec-transform-set-transform1] protocol esp
[DeviceB-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceB-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceB-ipsec-transform-set-transform1] quit
```

6. Configure an IKEv2 keychain to specify the key information used for IKEv2 communication.

```
[DeviceB]ikev2 keychain keychain1
[DeviceB-ikev2-keychain-keychain1] peer peer1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] address 3.3.3.1 16
[DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 3.3.3.1
[DeviceB-ikev2-keychain-keychain1-peer-peer1] pre-shared-key plaintext 123
```

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceB-ikev2-keychain-keychain1] quit
```

7. Configure an IKEv2 profile to specify the security parameters used for setting up IKEv2 SAs.

```
[DeviceB] ikev2 profile profile1
[DeviceB-ikev2-profile-profile1] keychain keychain1
[DeviceB-ikev2-profile-profile1] match remote identity fqdn www.devicea.com
[DeviceB-ikev2-profile-profile1] authentication-method local pre-share
[DeviceB-ikev2-profile-profile1] authentication-method remote pre-share
[DeviceB-ikev2-profile-profile1] quit
```

8. Configure an IPsec profile to establish an IPsec tunnel to protect the specified data:

```
[DeviceB] ipsec profile abc isakmp
[DeviceB-ipsec-profile-isakmp-abc] transform-set transform1
[DeviceB-ipsec-profile-isakmp-abc] ikev2-profile profile1
[DeviceB-ipsec-profile-isakmp-abc] quit
```

9. Configure an IPsec tunnel interface, which encapsulates the traffic to be protected with IPsec:

# Create an IPsec tunnel interface, and apply the IPsec profile.

```
[DeviceB] interface tunnel 1 mode ipsec
[DeviceB-Tunnel1] ip address 4.4.4.2 255.255.255.0
[DeviceB-Tunnel1] source 2.2.2.2
[DeviceB-Tunnel1] destination 3.3.3.1
[DeviceB-Tunnel1] tunnel protection ipsec profile abc
[DeviceB-Tunnel1] quit
```

# Add the IPsec tunnel interface to security zone **Untrust**.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface tunnel 1
[DeviceB-security-zone-Untrust] quit
```

# Configure a static route to direct the traffic to be protected to the IPsec tunnel interface.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 tunnel 1
```

Verifying the configuration

After the configuration is completed, Device A will automatically initiate IKE negotiation with Device B. After IKE negotiation succeeds, the tunnel interface will come up and traffic between two subnets will be IPsec-protected. This example uses Device A to verify the configuration.

# Display brief IP configuration for interfaces on Device A.

```
[DeviceA] display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface           Physical Protocol IP address/Mask    VPN instance Description
GE1/0/1             up       up       1.1.1.1/16         --           --
GE1/0/2             up       up       10.1.1.1/24        --           --
Tun1                up       up       4.4.4.1/24         --           --
```

# Display tunnel interface information on Device A.

```
[DeviceA] display interface Tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
```

```
Maximum transmission unit: 1444
Internet address: 4.4.4.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel TTL 255
Tunnel protocol/transport IPsec/IP
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: Tunnel1
-------------------------------
  -----------------------------
  IPsec profile: abc
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Transmitting entity: Initiator
    Path MTU: 1388
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.2
    Flow:
        sour addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
        dest addr: 0.0.0.0/0.0.0.0  port: 0  protocol: ip
    [Inbound ESP SAs]
      SPI: 2701952073 (0xa10c8449)
      Connection ID: 4294967296
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max received sequence-number: 0
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active
    [Outbound ESP SAs]
      SPI: 3607077598 (0xd6ffa2de)
      Connection ID: 12884901889
      Transform set: ESP-ENCRYPT-SM1-CBC-128 ESP-AUTH-SM3
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3180
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
[DeviceA] ping -a 10.1.1.2 10.1.2.2
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.2.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.000 ms
--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

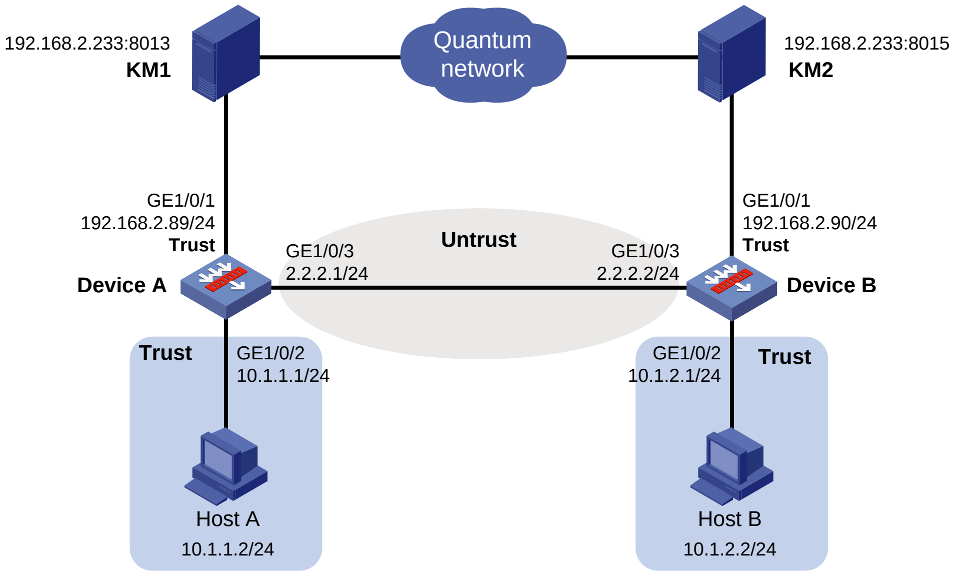
Example: Configuring IPsec GD-quantum encryption

Network configuration

Device A and Device B establish an IPsec tunnel through IKE negotiation and use quantum keys to encrypt the traffic between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Device A and Device B obtain the quantum keys for IPsec from GD-quantum servers KM 1 and KM 2, and use the default IKE proposal to negotiate IPsec SAs.

Figure 34 Network diagram



Analysis

Assume that GD-quantum servers KM1 and KM2 have been deployed successfully.

To establish an IPsec connection between Device A and Device B, finish the IKE and IPsec related configuration.

For Device A and Device B to obtain the quantum keys for IPsec, enable GD-quantum encryption on the devices, and connect Device A and Device B to KM 1 and KM2, respectively.

Restrictions and guidelines

As a best practice, deploy the devices that use quantum keys and the GD-quantum servers in the same LAN.

GD-quantum encryption is supported only by NSFOCUS devices. It cannot communicate with devices from other vendors.

GD-quantum encryption supports only IKEv1 in main mode.

GD-quantum encryption is not supported in non-default vSystems.

Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 192.168.2.89 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

   ```
   [DeviceA] ip route-static 10.1.2.0 24 2.2.2.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/3
   [DeviceA-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout1** to allow Device A to send IPsec negotiation packets to Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name ipseclocalout1
      [DeviceA-security-policy-ip-1-ipseclocalout1] source-zone local
      [DeviceA-security-policy-ip-1-ipseclocalout1] destination-zone untrust
      [DeviceA-security-policy-ip-1-ipseclocalout1] source-ip-host 2.2.2.1
      [DeviceA-security-policy-ip-1-ipseclocalout1] destination-ip-host 2.2.2.2
      [DeviceA-security-policy-ip-1-ipseclocalout1] action pass
      [DeviceA-security-policy-ip-1-ipseclocalout1] quit
      ```

      # Configure a rule named **ipseclocalin1** to allow Device A to receive the IPsec negotiation packets sent from Device B.

      ```
      [DeviceA-security-policy-ip] rule name ipseclocalin1
      [DeviceA-security-policy-ip-2-ipseclocalin1] source-zone untrust
      [DeviceA-security-policy-ip-2-ipseclocalin1] destination-zone local
      [DeviceA-security-policy-ip-2-ipseclocalin1] source-ip-host 2.2.2.2
      ```

```
[DeviceA-security-policy-ip-2-ipseclocalin1] destination-ip-host 2.2.2.1
[DeviceA-security-policy-ip-2-ipseclocalin1] action pass
[DeviceA-security-policy-ip-2-ipseclocalin1] quit
```

b. Configure rules to permit the traffic between Host A and Host B:

# Configure a rule named **trust-untrust** to permit the packets from Host A to Host B.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit the packets from Host B to Host A.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.2.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

c. Configure rules to permit traffic between Device A and KM 1.

# Configure a rule named **ipseclocalout2** to allow Device A to send packets to KM 1.

```
[DeviceA-security-policy-ip] rule name ipseclocalout2
[DeviceA-security-policy-ip-5-ipseclocalout2] source-zone local
[DeviceA-security-policy-ip-5-ipseclocalout2] destination-zone trust
[DeviceA-security-policy-ip-5-ipseclocalout2] source-ip-host 192.168.2.89
[DeviceA-security-policy-ip-5-ipseclocalout2] destination-ip-host 192.168.2.233
[DeviceA-security-policy-ip-5-ipseclocalout2] action pass
[DeviceA-security-policy-ip-5-ipseclocalout2] quit
```

# Configure a rule named **ipseclocalin2** to allow Device A to receive packets from KM 1.

```
[DeviceA-security-policy-ip] rule name ipseclocalin2
[DeviceA-security-policy-ip-6-ipseclocalin2] source-zone trust
[DeviceA-security-policy-ip-6-ipseclocalin2] destination-zone local
[DeviceA-security-policy-ip-6-ipseclocalin2] source-ip-host 192.168.2.233
[DeviceA-security-policy-ip-6-ipseclocalin2] destination-ip-host 192.168.2.89
[DeviceA-security-policy-ip-6-ipseclocalin2] action pass
[DeviceA-security-policy-ip-6-ipseclocalin2] quit
[DeviceA-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3000 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
[DeviceA] acl advanced 3001
[DeviceA-acl-ipv4-adv-3001] rule permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3001] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings used by both sides of the tunnel must be the same.

```
[DeviceA] ipsec transform-set tran1
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceA-ipsec-transform-set-tran1] protocol esp
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceA-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

   The preshared key used by both sides of the communication must be the same.

   ```
   [DeviceA] ike keychain keychain1
   [DeviceA-ike-keychain-keychain1] pre-shared-key address 2.2.2.2 255.255.255.0 key
   simple 123456TESTplat&!
   [DeviceA-ike-keychain-keychain1] quit
   ```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

   ```
   [DeviceA] ike profile profile1
   [DeviceA-ike-profile-profile1] keychain keychain1
   [DeviceA-ike-profile-profile1] local-identity address 2.2.2.1
   [DeviceA-ike-profile-profile1] match remote identity address 2.2.2.2 255.255.255.0
   [DeviceA-ike-profile-profile1] quit
   ```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry, which specifies the ACL for IPsec, IPsec transform set, local and remote IP addresses, and IKE profile.

   ```
   [DeviceA] ipsec policy map1 10 isakmp
   [DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
   [DeviceA-ipsec-policy-isakmp-map1-10] security acl 3001
   [DeviceA-ipsec-policy-isakmp-map1-10] transform-set tran1
   [DeviceA-ipsec-policy-isakmp-map1-10] ike-profile profile1
   [DeviceA-ipsec-policy-isakmp-map1-10] quit
   ```

10. Apply the IPsec policy to GigabitEthernet 1/0/3 to protect traffic on the interface.

    ```
    [DeviceA] interface gigabitethernet 1/0/3
    [DeviceA-GigabitEthernet1/0/3] ipsec apply policy map1
    [DeviceA-GigabitEthernet1/0/3] quit
    ```

11. Enable GD-quantum encryption, and configure related parameters.

    ```
    [DeviceA] ike gd-quantum
    [DeviceA-ike-gdquantum] app-dev-info 341300002
    [DeviceA-ike-gdquantum] server-address 192.168.2.233 port 8013
    [DeviceA-ike-gdquantum] auth-key simple
    66c7f0f462eeedd9d1f2d46bdc10e4e24167c4875cf2f7a2297da02b8f4ba8e0
    [DeviceA-ike-gdquantum] decrypt-quantum-key simple
    66c7f0f462eeedd9d1f2d46bdc10e4e24167c4875cf2f7a2297da02b8f4ba8e0
    [DeviceA-ike-gdquantum] quit
    ```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 192.168.2.90 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.1.

   ```
   [DeviceB] ip route-static 10.1.1.0 24 2.2.2.1
   ```

3. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/3
   [DeviceB-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   a. Configure rules to permit traffic between the **Untrust** and **Local** security zones, so the devices can set up an IPsec tunnel:

      # Configure a rule named **ipseclocalout1** to allow Device B to send IPsec negotiation packets to Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name ipseclocalout1
      [DeviceB-security-policy-ip-1-ipseclocalout1] source-zone local
      [DeviceB-security-policy-ip-1-ipseclocalout1] destination-zone untrust
      [DeviceB-security-policy-ip-1-ipseclocalout1] source-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-1-ipseclocalout1] destination-ip-host 2.2.2.1
      [DeviceB-security-policy-ip-1-ipseclocalout1] action pass
      [DeviceB-security-policy-ip-1-ipseclocalout1] quit
      ```

      # Configure a rule named **ipseclocalin1** to allow Device B to receive the IPsec negotiation packets sent from Device A.

      ```
      [DeviceB-security-policy-ip] rule name ipseclocalin1
      [DeviceB-security-policy-ip-2-ipseclocalin1] source-zone untrust
      [DeviceB-security-policy-ip-2-ipseclocalin1] destination-zone local
      [DeviceB-security-policy-ip-2-ipseclocalin1] source-ip-host 2.2.2.1
      [DeviceB-security-policy-ip-2-ipseclocalin1] destination-ip-host 2.2.2.2
      [DeviceB-security-policy-ip-2-ipseclocalin1] action pass
      [DeviceB-security-policy-ip-2-ipseclocalin1] quit
      ```

   b. Configure rules to permit traffic between Host B and Host A:

      # Configure a rule named **trust-untrust** to permit the packets from Host B to Host A.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.2.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] destination-ip-subnet 10.1.1.0 24
      [DeviceB-security-policy-ip-3-trust-untrust] action pass
      [DeviceB-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit the packets from Host A to Host B.

      ```
      [DeviceB-security-policy-ip] rule name untrust-trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
      [DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
      [DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.1.0 24
      ```

```
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

c. Configure rules to permit traffic between Device B and KM 2.

# Configure a rule named **ipseclocalout2** to allow Device B to send packets to KM 2.

```
[DeviceB-security-policy-ip] rule name ipseclocalout2
[DeviceB-security-policy-ip-5-ipseclocalout2] source-zone local
[DeviceB-security-policy-ip-5-ipseclocalout2] destination-zone trust
[DeviceB-security-policy-ip-5-ipseclocalout2] source-ip-host 192.168.2.90
[DeviceB-security-policy-ip-5-ipseclocalout2] destination-ip-host 192.168.2.233
[DeviceB-security-policy-ip-5-ipseclocalout2] action pass
[DeviceB-security-policy-ip-5-ipseclocalout2] quit
```

# Configure a rule named **ipseclocalin2** to allow Device B to receive packets from KM 2.

```
[DeviceB-security-policy-ip] rule name ipseclocalin2
[DeviceB-security-policy-ip-6-ipseclocalin2] source-zone trust
[DeviceB-security-policy-ip-6-ipseclocalin2] destination-zone local
[DeviceB-security-policy-ip-6-ipseclocalin2] source-ip-host 192.168.2.233
[DeviceB-security-policy-ip-6-ipseclocalin2] destination-ip-host 192.168.2.90
[DeviceB-security-policy-ip-6-ipseclocalin2] action pass
[DeviceB-security-policy-ip-6-ipseclocalin2] quit
[DeviceB-security-policy-ip] quit
```

5. Define the data flows to be protected:

# Configure IPv4 advanced ACL 3001 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
[DeviceB] acl advanced 3001
[DeviceB-acl-ipv4-adv-3001] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3001] quit
```

6. Configure an IPsec transform set to specify the packet encapsulation mode, security protocols, and algorithms.

The IPsec transform set settings used by both sides of the tunnel must be the same.

```
[DeviceB] ipsec transform-set tran1
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
[DeviceB-ipsec-transform-set-tran1] protocol esp
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm aes-cbc-128
[DeviceB-ipsec-transform-set-tran1] esp authentication-algorithm sha1
[DeviceB-ipsec-transform-set-tran1] quit
```

7. Configure an IKE keychain to specify the key information used for IKE communication.

The preshared key used by both sides of the communication must be the same.

```
[DeviceB]ike keychain keychain1
[DeviceB-ike-keychain-keychain1] pre-shared-key address 2.2.2.1 255.255.255.0 key
simple 123456TESTplat&!
[DeviceB-ike-keychain-keychain1] quit
```

8. Configure an IKE profile to specify the security parameters used for setting up IKE SAs.

```
[DeviceB] ike profile profile1
[DeviceB-ike-profile-profile1] keychain keychain1
[DeviceB-ike-profile-profile1] local-identity address 2.2.2.2
```

```
[DeviceB-ike-profile-profile1] match remote identity address 2.2.2.1 255.255.255.0
[DeviceB-ike-profile-profile1] quit
```

9. Configure an IPsec policy, so as to establish the IPsec tunnel to protect data:

   # Create an IKE-based IPsec policy entry.

   ```
   [DeviceB] ipsec policy use1 10 isakmp
   [DeviceB-ipsec-policy-isakmp-use1-10] remote-address 2.2.2.1
   [DeviceB-ipsec-policy-isakmp-use1-10] security acl 3001
   [DeviceB-ipsec-policy-isakmp-use1-10] transform-set tran1
   [DeviceB-ipsec-policy-isakmp-use1-10] ike-profile profile1
   [DeviceB-ipsec-policy-isakmp-use1-10] quit
   ```

10. Apply the IPsec policy to GigabitEthernet 1/0/3 to protect traffic on the interface.

    ```
    [DeviceB] interface gigabitethernet 1/0/3
    [DeviceB-GigabitEthernet1/0/3] ipsec apply policy use1
    [DeviceB-GigabitEthernet1/0/3] quit
    ```

11. Enable GD-quantum encryption, and configure related parameters.

    ```
    [DeviceB] ike gd-quantum
    [DeviceB-ike-gdquantum] app-dev-info 341300001
    [DeviceB-ike-gdquantum] server-address 192.168.2.233 port 8015
    [DeviceB-ike-gdquantum] auth-key simple
    66c7f0f462eeedd9d1f2d46bdc10e4e24167c4875cf2f7a2297da02b8f4ba8e0
    [DeviceB-ike-gdquantum] decrypt-quantum-key simple
    66c7f0f462eeedd9d1f2d46bdc10e4e24167c4875cf2f7a2297da02b8f4ba8e0
    [DeviceB-ike-gdquantum] quit
    ```

Verifying the configuration

# Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKE negotiation. After IKE SAs are successfully negotiated, Device A and Device B request an encrypted quantum key from KM 1 and KM 2, respectively. They decrypt the quantum key by using the configured decryption key to get the quantum key for IPsec.

# Verify that a private IP address in subnet 10.1.1.0/24 can ping a private IP address in subnet 10.1.2.0/24 successfully.

```
<Device A>ping -a 10.1.1.1 10.1.2.1
Ping 10.1.2.1 (10.1.2.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
Request time out
56 bytes from 10.1.2.1: icmp_seq=1 ttl=255 time=0.652 ms
56 bytes from 10.1.2.1: icmp_seq=2 ttl=255 time=0.294 ms
56 bytes from 10.1.2.1: icmp_seq=3 ttl=255 time=0.244 ms
56 bytes from 10.1.2.1: icmp_seq=4 ttl=255 time=0.237 ms

--- Ping statistics for 10.1.2.1 ---
5 packet(s) transmitted, 4 packet(s) received, 20.0% packet loss
round-trip min/avg/max/std-dev = 0.237/0.357/0.652/0.172 ms
```

# Display the IKE proposal configuration on Device A and Device B. The default IKE proposal is displayed.

```
[DeviceA] display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
              method       algorithm    algorithm       group      (seconds)
----------------------------------------------------------------------------
default  PRE-SHARED-KEY     SHA1         DES-CBC        Group 1      86400
```

```
[DeviceB] display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
              method       algorithm    algorithm       group      (seconds)
----------------------------------------------------------------------------
default  PRE-SHARED-KEY     SHA1         DES-CBC        Group 1      86400
```

# Display IKE SAs established in phase-1 IKE negotiation.

```
[DeviceA] display ike sa
    Connection-ID  Local               Remote                Flag     DOI
---------------------------------------------------------------------------------
    1              2.2.2.1           2.2.2.2/500         RD       IPsec
Flags:
RD--READY RL--REPLACED FD-FADING RK-REKEY
```

# Display IPsec SAs established in phase-2 IKE negotiation.

```
[DeviceA] display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/3
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Alias: map1-10
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect Forward Secrecy:
    Inside VPN:
    Extended Sequence Numbers enable: N
    Traffic Flow Confidentiality enable: N
    Transmitting entity: Initiator
    Path MTU: 1428
    Tunnel:
        local  address/port: 2.2.2.1/500
        remote address/port: 2.2.2.2/500
    Flow:
        sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
        dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 117833058 (0x0705fd62)
      Connection ID: 150323855360
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3247
      Max received sequence-number: 10
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: Active

    [Outbound ESP SAs]
      SPI: 3302519622 (0xc4d87346)
      Connection ID: 47244640257
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3247
      Max sent sequence-number: 10
      UDP encapsulation used for NAT traversal: N
      Status: Active
```

On Device B, display the established IKE SAs and IPsec SAs in the same way. (Details not shown.)

Configuring IKE

Unless otherwise specified, the term "IKE" in this chapter refers to IKEv1.

About IKE

Built on a framework defined by ISAKMP, Internet Key Exchange (IKE) provides automatic key negotiation and SA establishment services for IPsec.

Benefits of IKE

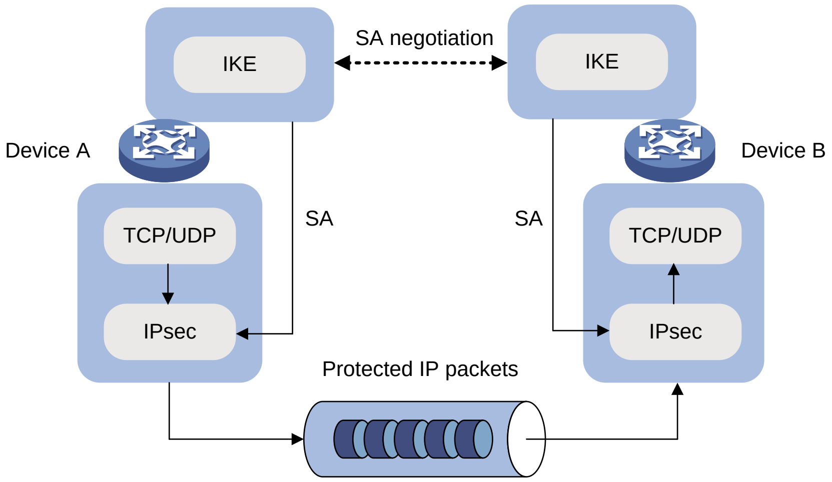
IKE provides the following benefits for IPsec:

- Automatically negotiates IPsec parameters.
- Performs DH exchanges to calculate shared keys, making sure each SA has a key that is independent of other keys.
- Automatically negotiates SAs when the sequence number in the AH or ESP header overflows, making sure IPsec can provide the anti-replay service by using the sequence number.

Relationship between IPsec and IKE

As shown in Figure 35, IKE negotiates SAs for IPsec and transfers the SAs to IPsec, and IPsec uses the SAs to protect IP packets.

Figure 35 Relationship between IKE and IPsec



IKE negotiation process

IKE negotiates keys and SAs for IPsec in two phases:

1. **Phase 1**—The two peers establish an IKE SA, a secure, authenticated channel for communication.
2. **Phase 2**—Using the IKE SA established in phase 1, the two peers negotiate to establish IPsec SAs.

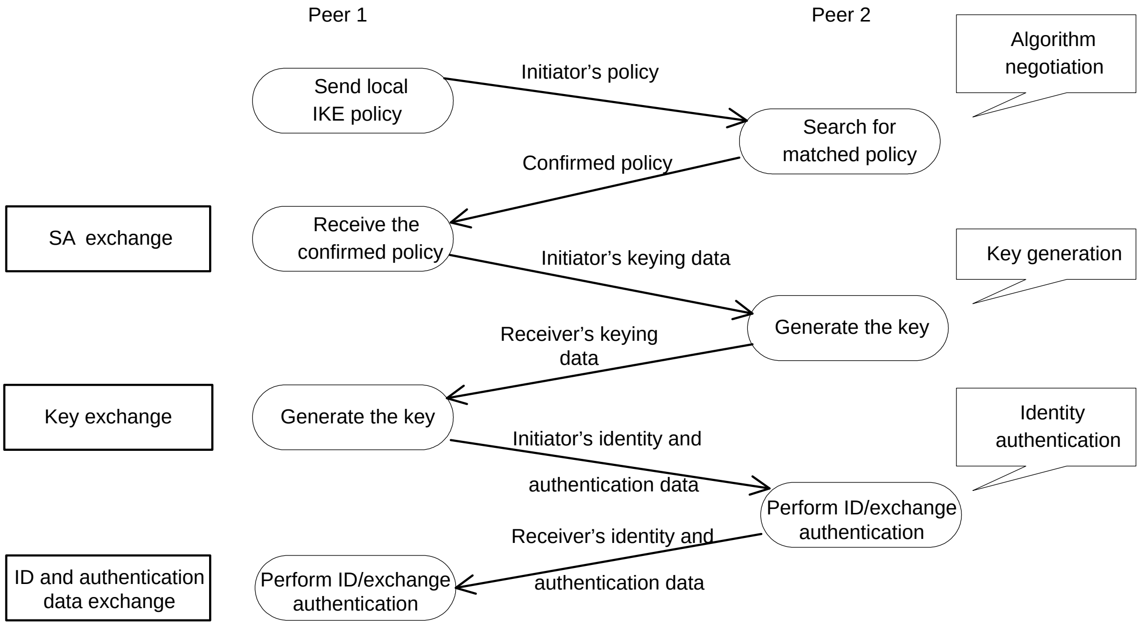
Phase 1 negotiation can use the main mode, GM main mode, or aggressive mode.

IKE exchange process in main mode

As shown in Figure 36, the main mode of IKE negotiation in phase 1 involves three pairs of messages:

- **SA exchange**—Used for negotiating the IKE security policy.
- **Key exchange**—Used for exchanging the DH public value and other values, such as the random number. The two peers use the exchanged data to generate key data and use the encryption key and authentication key to ensure the security of IP packets.
- **ID and authentication data exchange**—Used for identity authentication.

Figure 36 IKE exchange process in main mode


Peer 1
Peer 2
Algorithm negotiation
Send local IKE policy
Initiator's policy
Search for matched policy
Confirmed policy
SA exchange
Receive the confirmed policy
Initiator's keying data
Key generation
Generate the key
Receiver's keying data
Key exchange
Generate the key
Initiator's identity and authentication data
Identity authentication
Perform ID/exchange authentication
Receiver's identity and authentication data
ID and authentication data exchange
Perform ID/exchange authentication

IKE exchange process in aggressive mode

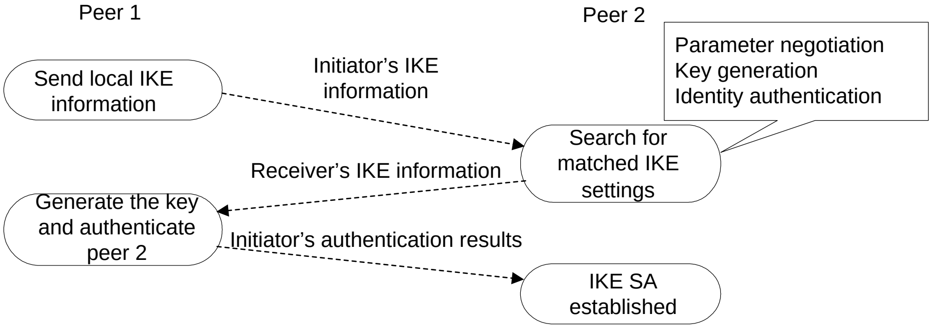
As shown in Figure 37, the process of phase 1 IKE negotiation in aggressive mode is as follows:

1. The initiator (peer 1) sends a message containing the local IKE information to peer 2. The message includes parameters used for IKE SA establishment, keying data, and peer 1's identity information.
2. Peer 2 chooses the IKE establishment parameters to use, generate the key, and authenticate peer 1's identity. Then it sends the IKE data to peer 1.
3. Peer 1 generates the key, authenticates peer 2's identity, and sends the results to peer 1.

After the preceding process, an IKE SA is established between peer 1 and peer 2.

The aggressive mode is faster than the main mode but it does not provide identity information protection. The main mode provides identity information protection but is slower. Choose the appropriate negotiation mode according to your requirements.

Figure 37 IKE exchange process in aggressive mode


Peer 1
Peer 2
Parameter negotiation
Key generation
Identity authentication
Send local IKE information
Initiator's IKE information
Search for matched IKE settings
Receiver's IKE information
Generate the key and authenticate peer 2
Initiator's authentication results
IKE SA established

IKE exchange process in GM main mode

If the local end uses the GM main mode, the two IKE peers must use the RSA-DE or SM2-DE digital envelop authentication method.

IKE security mechanism

IKE has a series of self-protection mechanisms and supports secure identity authentication, key distribution, and IPsec SA establishment on insecure networks.

Identity authentication

The IKE identity authentication mechanism is used to authenticate the identity of the communicating peers. The device supports the following identity authentication methods:

- **Preshared key authentication**—Two communicating peers use the pre-configured shared key for identity authentication.
- **RSA signature authentication** and **DSA signature authentication**—Two communicating peers use the digital certificates issued by the CA for identity authentication.

The preshared key authentication method does not require certificates and is easy to configure. It is usually deployed in small networks.

The signature authentication methods provide higher security and are usually deployed in networks with the headquarters and some branches. When deployed in a network with many branches, a signature authentication method can simplify the configuration because only one PKI domain is required. If you use the preshared key authentication method, you must configure a preshared key for each branch on the Headquarters node.

DH algorithm

The DH algorithm is a public key algorithm. With this algorithm, two peers can exchange keying material and then use the material to calculate the shared keys. Due to the decryption complexity, a third party cannot decrypt the keys even after intercepting all keying materials.

PFS

The Perfect Forward Secrecy (PFS) feature is a security feature based on the DH algorithm. After PFS is enabled, an additional DH exchange is performed in IKE phase 2 to make sure IPsec keys have no derivative relations with IKE keys and a broken key brings no threats to other keys.

Protocols and standards

- RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*
- RFC 2409, *The Internet Key Exchange (IKE)*
- RFC 2412, *The OAKLEY Key Determination Protocol*

- Internet Draft, *draft-ietf-ipsec-isakmp-xauth-06*

IKE tasks at a glance

To configure IKE, perform the following tasks:

1. (Optional.) Configuring an IKE profile
   a. Creating an IKE profile
   b. Configuring peer IDs for the IKE profile
   c. Specifying the IKE keychain or PKI domain
   d. Configuring the IKE phase 1 negotiation mode
   e. Specifying IKE proposals for the IKE profile
   f. Configuring the local ID for the IKE profile
   g. Specifying an inside VPN instance for the IKE profile
   h. Configuring optional features for the IKE profile
2. Configuring an IKE proposal
3. Configuring an IKE keychain
4. (Optional.) Configuring accessibility features for IKE negotiation
   - Configuring the global identity information
   - Configuring the IKE keepalive feature
   - Configuring the IKE NAT keepalive feature
   - Configuring global IKE DPD
   - Enabling invalid SPI recovery
   - Setting the maximum number of IKE SAs
   - Configuring IKE address pools
   - Configuring IKE negotiation compatibility
   - Configuring GD-quantum encryption
   - Configuring SNMP notifications and logging for IKE

Prerequisites for IKE configuration

Determine the following parameters prior to IKE configuration:

- The algorithms to be used during IKE negotiation, including the identity authentication method, encryption algorithm, authentication algorithm, and DH group.
  - Different algorithms provide different levels of protection. A stronger algorithm provides more resistance to decryption but uses more resources.
  - A DH group that uses more bits provides higher security but needs more time for processing.
- The preshared key or PKI domain for IKE negotiation. For more information about PKI, see "Configuring PKI."
- The IKE-based IPsec policies for the communicating peers. If you do not specify an IKE profile in an IPsec policy, the device selects an IKE profile for the IPsec policy. If no IKE profile is configured, the globally configured IKE settings are used. For more information about IPsec, see "Configuring IPsec."

Configuring an IKE profile

Creating an IKE profile

About this task

Perform this task to create an IKE profile.

An IKE profile is intended to provide a set of parameters for IKE negotiation.

Procedure

1. Enter system view.
   system-view
2. Create an IKE profile and enter its view.
   ike profile *profile-name*

Configuring peer IDs for the IKE profile

About this task

Perform this task to configure the peer IDs for IKE profile matching. When the device needs to select an IKE profile for IKE negotiation with a peer, it compares the received peer ID with the peer IDs of its local IKE profiles. If a match is found, it uses the IKE profile with the matching peer ID for IKE negotiation.

Restrictions and guidelines

For an IKE profile, you can configure multiple peer IDs. A peer ID configured earlier has a higher priority.

Two IKE peers must both have or both not have peer IDs configured.

Procedure

1. Enter system view.
   system-view
2. Enter IKE profile view.
   ike profile *profile-name*
3. Configure a peer ID for the IKE profile.
   match remote { **certificate** *policy-name* | **identity** { **address** { { *ipv4-address* [*mask* | *mask-length*] | **range** *low-ipv4-address high-ipv4-address* } | **ipv6** { *ipv6-address* [*prefix-length*] | **range** *low-ipv6-address high-ipv6-address* } } [**vpn-instance** *vpn-instance-name*] | **fqdn** *fqdn-name* | **user-fqdn** *user-fqdn-name* } }

Specifying the IKE keychain or PKI domain

Restrictions and guidelines

Configure the IKE keychain or PKI domain for the IKE proposals to use. To use digital signature authentication, configure a PKI domain. To use preshared key authentication, configure an IKE keychain.

Procedure

1. Enter system view.

system-view

2. Enter IKE profile view.

   ike profile *profile-name*

3. Specify the keychain for preshared key authentication or the PKI domain used to request a certificate for digital signature authentication.

   - Specify the keychain.

     keychain *keychain-name*

   - Specify the PKI domain.

     certificate domain *domain-name*

   By default, no IKE keychain or PKI domain is specified in an IKE profile.

Configuring the IKE phase 1 negotiation mode

Restrictions and guidelines

Specify the IKE phase 1 negotiation mode (main or aggressive) that the device uses as the initiator. When the device acts as the responder, it uses the IKE negotiation mode of the initiator.

Procedure

1. Enter system view.

   system-view

2. Enter IKE profile view.

   ike profile *profile-name*

3. Specify the IKE negotiation mode for phase 1.

   exchange-mode { **aggressive** | **gm-main** | **main** }

   By default, IKE negotiation in phase 1 uses the main mode.

Specifying IKE proposals for the IKE profile

Restrictions and guidelines

Specify the IKE proposals that the device can use as the initiator. An IKE proposal specified earlier has a higher priority. When the device acts as the responder, it uses the IKE proposals configured in system view to match the IKE proposals received from the initiator. If no matching proposal is found, the negotiation fails.

Procedure

1. Enter system view.

   system-view

2. Enter IKE profile view.

   ike profile *profile-name*

3. Specify IKE proposals for the IKE profile.

   proposal *proposal-number*&<1-6>

   By default, no IKE proposals are specified for an IKE profile and the IKE proposals configured in system view are used for IKE negotiation.

Configuring the local ID for the IKE profile

Restrictions and guidelines

For digital signature authentication, the device can use an ID of any type. If the local ID is an IP address that is different from the IP address in the local certificate, the device uses the FQDN (the device name configured by using the **sysname** command) instead.

For preshared key authentication, the device can use an ID of any type other than the DN.

Procedure

1. Enter system view.

   system-view

2. Enter IKE profile view.

   ike profile *profile-name*

3. Configure the local ID.

   local-identity { **address** { *ipv4-address* | **ipv6** *ipv6-address* } | **dn** | **fqdn** [*fqdn-name*] | **user-fqdn** [*user-fqdn-name*] }

   By default, no local ID is configured for an IKE profile, and an IKE profile uses the local ID configured in system view. If the local ID is not configured in system view, the IKE profile uses the IP address of the interface to which the IPsec policy or IPsec policy template is applied as the local ID.

Specifying an inside VPN instance for the IKE profile

About this task

The inside MPLS L3VPN instance determines where the device should forward received IPsec protected data. If you specify an inside VPN instance, the device looks for a route in the specified VPN instance to forward the data. If you do not specify an inside VPN instance, the device looks for a route in the VPN instance where the receiving interface resides to forward the data.

Restrictions and guidelines

The inside VPN instance specified in an IKE profile takes effect only on IPsec policies that use the IKE profile. It does not take effect on IPsec profiles that use the IKE profile.

Procedure

1. Enter system view.

   system-view

2. Enter IKE profile view.

   ike profile *profile-name*

3. Specify an inside VPN instance.

   inside-vpn vpn-instance *vpn-instance-name*

   By default, no inside VPN instance is specified for an IKE profile, and the device forwards protected data to the VPN instance where the interface receiving the data resides.

Configuring optional features for the IKE profile

1. Enter system view.

   system-view

2. Enter IKE profile view.

ike profile *profile-name*

3. Configure optional features as needed.
   - Configure IKE DPD.

     dpd interval *interval* [**retry** *seconds*] { **on-demand** | **periodic** }

     By default, IKE DPD is not configured for an IKE profile and an IKE profile uses the DPD settings configured in system view. If IKE DPD is not configured in system view either, the device does not perform dead IKE peer detection.

     The IKE DPD settings configured in the IKE profile view take precedence over those configured in system view.
   - Specify the local interface or IP address to which the IKE profile can be applied.

     match local address { *interface-type interface-number* | { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] }

     By default, an IKE profile can be applied to any local interface or IP address.

     An IKE profile configured with this command has a higher priority over those not configured with this command.
   - Specify a priority for the IKE profile.

     priority *priority*

     By default, the priority of an IKE profile is 100.

     The device selects a local IKE profile for IKE negotiation as follows:
     - First, it selects an IKE profile with the **match local address** command configured.
     - If a tie exists, it selects the IKE profile with a smaller priority number.
     - If a tie still exists, it selects the IKE profile configured earlier.
   - Enable client authentication.

     client-authentication xauth

     By default, client authentication is disabled.

     Configure this command in the IKE profile used by the IPsec gateway at the enterprise center. This command enables the IPsec gateway to perform extended (XAUTH) authentication on remote users through AAA after IKE phase 1 negotiation. AAA configuration is also required on the IPsec gateway for client authentication.
   - Specify the username and password for client authentication.

     client-authentication xauth user

     By default, the username and password for client authentication are not specified.

     Configure this command in the IKE profile used by a branch gateway. The branch gateway can then use the username and password to pass AAA authentication and establish an IPsec tunnel with the IPsec gateway at the enterprise center.
   - Enable AAA authorization.

     aaa authorization domain *domain-name* **username** *user-name*

     By default, AAA authorization is disabled.

     The AAA authorization feature enables IKE to request authorization attributes, such as the IKE address pool, from AAA. IKE uses the address pool to assign IP addresses to remote users. For more information about AAA authorization, see "Configuring AAA."
   - (Optional.) Set the IKE SA soft lifetime buffer time.

     sa soft-duration buffer *seconds*

     By default, the IKE SA soft lifetime buffer time is not configured.

Set the IKE SA soft lifetime buffer time to determine the IKE SA soft lifetime. The SA soft lifetime is calculated as follows: SA soft lifetime = SA lifetime – SA soft lifetime buffer. A new IKE SA will be negotiated when the SA soft lifetime timer expires.

- Specify the version of the SM4 algorithm used for IKE GM main negotiations.

  ike gm-main sm4-version { **draft** | **standard** }

  By default, the standard SM4 algorithm is used in IKE GM main negotiations.

  Specify the SM4 version used by the device to initiate an IKE negotiation with a device from other vendors to make sure the two devices use the same SM4 version in the negotiation.

Configuring an IKE proposal

About this task

An IKE proposal defines a set of attributes describing how IKE negotiation in phase 1 should take place. You can create multiple IKE proposals with different priorities. The priority of an IKE proposal is represented by its sequence number. The lower the sequence number, the higher the priority.

Two peers must have at least one matching IKE proposal for successful IKE negotiation. During IKE negotiation:

- The initiator sends its IKE proposals to the peer.
  - If the initiator is using an IPsec policy with an IKE profile, the initiator sends all IKE proposals specified in the IKE profile to the peer. An IKE proposal specified earlier for the IKE profile has a higher priority.
  - If the initiator is using an IPsec policy with no IKE profile, the initiator sends all its IKE proposals to the peer. An IKE proposal with a smaller number has a higher priority.
- The peer searches its own IKE proposals for a match. The search starts from the IKE proposal with the highest priority and proceeds in descending order of priority until a match is found. The matching IKE proposals are used to establish the IKE SA. If all user-defined IKE proposals are found mismatching, the two peers use their default IKE proposals to establish the IKE SA.

Two matching IKE proposals have the same encryption algorithm, authentication method, authentication algorithm, and DH group. The SA lifetime takes the smaller one of the two proposals' SA lifetime settings.

Procedure

1. Enter system view.

   system-view

2. Create an IKE proposal and enter its view.

   ike proposal *proposal-number*

   By default, a default IKE proposal exists.

3. Configure a description for the IKE proposal.

   description

   By default, an IKE proposal does not have a description.

4. Specify an encryption algorithm for the IKE proposal.

   encryption-algorithm { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **des-cbc** | **sm1-cbc-128** | **sm4-cbc** }

   By default, the 56-bit DES encryption algorithm in CBC mode is used.

5. Specify an authentication method for the IKE proposal.

   authentication-method { **dsa-signature** | **pre-share** | **rsa-de** | **rsa-signature** | **sm2-de** }

   By default, the preshared key authentication method is used.

6. Specify an authentication algorithm for the IKE proposal.

   authentication-algorithm { **md5** | **sha** | **sha256** | **sha384** | **sha512** | **sm3** }

   By default, the HMAC-SHA1 authentication algorithm is used.

7. Specify a DH group for key negotiation in phase 1.

   dh { **group1** | **group14** | **group2** | **group24** | **group5** }

   DH group 1 (the 768-bit DH group) is used by default.

8. (Optional.) Set the IKE SA lifetime for the IKE proposal.

   sa duration *seconds*

   By default, the IKE SA lifetime is 86400 seconds.

   If the IPsec SA lifetime is also configured, set the IKE SA lifetime longer than the IPsec SA lifetime as a best practice.

Configuring an IKE keychain

About this task

Perform this task when you configure the IKE to use the preshared key for authentication.

Follow these guidelines when you configure an IKE keychain:

- Two peers must be configured with the same preshared key to pass preshared key authentication.
- You can specify the local address configured in IPsec policy or IPsec policy template view (using the **local-address** command) for the IKE keychain to be applied. If no local address is configured, specify the IP address of the interface that uses the IPsec policy.
- The device determines the priority of an IKE keychain as follows:
  a. The device examines the existence of the **match local address** command. An IKE keychain with the **match local address** command configured has a higher priority.
  b. If a tie exists, the device compares the priority numbers. An IKE keychain with a smaller priority number has a higher priority.
  c. If a tie still exists, the device prefers an IKE keychain configured earlier.

Procedure

1. Enter system view.

   system-view

2. Create an IKE keychain and enter its view.

   ike keychain *keychain-name* [**vpn-instance** *vpn-instance-name*]

3. Configure a preshared key.

   pre-shared-key { **address** { *ipv4-address* [*mask* | *mask-length*] | **ipv6** *ipv6-address* [*prefix-length*] } | **hostname** *host-name* } **key** { **cipher** | **simple** } *string*

   By default, no preshared key is configured.

4. (Optional.) Specify a local interface or IP address to which the IKE keychain can be applied.

   match local address { *interface-type interface-number* | { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] }

   By default, an IKE keychain can be applied to any local interface or IP address.

5. (Optional.) Specify a priority for the IKE keychain.

   priority *priority*

   The default priority is 100.

Configuring the global identity information

Restrictions and guidelines

The global identity can be used by the device for all IKE SA negotiations, and the local identity (set by the **`local-identity`** command) can be used only by the device that uses the IKE profile.

When signature authentication is used, you can set any type of the identity information.

When preshared key authentication is used, you cannot set the DN as the identity.

Procedure

1. Enter system view.

   `system-view`

2. Configure the global identity to be used by the local end.

   `ike identity` { **`address`** { *`ipv4-address`* | **`ipv6`** *`ipv6-address`* }| **`dn`** | **`fqdn`** [*`fqdn-name`*] | **`user-fqdn`** [*`user-fqdn-name`*] }

   By default, the IP address of the interface to which the IPsec policy or IPsec policy template is applied is used as the IKE identity.

3. (Optional.) Configure the local device to always obtain the identity information from the local certificate for signature authentication.

   `ike signature-identity from-certificate`

   By default, the local end uses the identity information specified by **`local-identity`** or **`ike identity`** for signature authentication.

   Configure this command when the aggressive mode and signature authentication are used and the device interconnects with a NF 5-based peer device. NF supports only DN for signature authentication.

Configuring the IKE keepalive feature

About this task

IKE sends keepalive packets to query the liveness of the peer. If the peer is configured with the keepalive timeout time, you must configure the keepalive interval on the local device. If the peer receives no keepalive packets during the timeout time, the IKE SA is deleted along with the IPsec SAs it negotiated.

Restrictions and guidelines

Configure IKE DPD instead of IKE keepalive unless IKE DPD is not supported on the peer. The IKE keepalive feature sends keepalives at regular intervals, which consumes network bandwidth and resources.

The keepalive timeout time configured on the local device must be longer than the keepalive interval configured at the peer. Since it seldom occurs that more than three consecutive packets are lost on a network, you can set the keepalive timeout three times as long as the keepalive interval.

Procedure

1. Enter system view.

   `system-view`

2. Set the IKE SA keepalive interval.

   `ike keepalive interval`*`interval`*

   By default, no keepalives are sent to the peer.

3. Set the IKE SA keepalive timeout time.

```
ike keepalive timeout seconds
```

By default, IKE SA keepalive never times out.

Configuring the IKE NAT keepalive feature

About this task

If IPsec traffic passes through a NAT device, you must configure the NAT traversal feature. If no packet travels across an IPsec tunnel in a period of time, the NAT sessions are aged and deleted, disabling the tunnel from transmitting data to the intended end. To prevent NAT sessions from being aged, configure the NAT keepalive feature on the IKE gateway behind the NAT device to send NAT keepalive packets to its peer periodically to keep the NAT session alive.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the IKE NAT keepalive interval.

   ```
   ike nat-keepalive seconds
   ```

   The default interval is 20 seconds.

Configuring global IKE DPD

About this task

DPD detects dead peers. It can operate in periodic mode or on-demand mode.

- **Periodic DPD**—Sends a DPD message at regular intervals. It features an earlier detection of dead peers, but consumes more bandwidth and CPU.
- **On-demand DPD**—Sends a DPD message based on traffic. When the device has traffic to send and is not aware of the liveness of the peer, it sends a DPD message to query the status of the peer. If the device has no traffic to send, it never sends DPD messages. As a best practice, use the on-demand mode.

The IKE DPD works as follows:

1. The local device sends a DPD message to the peer, and waits for a response from the peer.
2. If the peer does not respond within the retry interval specified by the **retry** *seconds* parameter, the local device resends the message.
3. If still no response is received within the retry interval, the local end sends the DPD message again. The system allows a maximum of four retries.
4. If the local device receives no response after four retries, the device considers the peer to be dead, and deletes the IKE SA along with the IPsec SAs it negotiated.
5. If the local device receives a response from the peer during the detection process, the peer is considered alive. The local device performs a DPD detection again when the triggering interval is reached or it has traffic to send, depending on the DPD mode.

Restrictions and guidelines

When DPD settings are configured in both IKE profile view and system view, the DPD settings in IKE profile view apply. If DPD is not configured in IKE profile view, the DPD settings in system view apply.

It is a good practice to set the triggering interval longer than the retry interval so that a DPD detection is not triggered during a DPD retry.

Procedure

1. Enter system view.

```
system-view
```

2. Enable sending IKE DPD messages.

   ike dpd interval *interval* [**retry** *seconds*] { **on-demand** | **periodic** }

   By default, IKE DPD is disabled.

Enabling invalid SPI recovery

About this task

An IPsec "black hole" occurs when one IPsec peer fails (for example, a peer can fail if a reboot occurs). One peer fails and loses its SAs with the other peer. When an IPsec peer receives a data packet for which it cannot find an SA, an invalid SPI is encountered. The peer drops the data packet and tries to send an SPI invalid notification to the data originator. This notification is sent by using the IKE SA. Because no IKE SA is available, the notification is not sent. The originating peer continues sending the data by using the IPsec SA that has the invalid SPI, and the receiving peer keeps dropping the traffic.

The invalid SPI recovery feature enables the receiving peer to set up an IKE SA with the originator so that an SPI invalid notification can be sent. Upon receiving the notification, the originating peer deletes the IPsec SA that has the invalid SPI. If the originator has data to send, new SAs will be set up.

Restrictions and guidelines

Use caution when you enable the invalid SPI recovery feature because using this feature can result in a DoS attack. Attackers can make a great number of invalid SPI notifications to the same peer.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable invalid SPI recovery.

   ```
   ike invalid-spi-recovery enable
   ```

   By default, the invalid SPI recovery is disabled.

Setting the maximum number of IKE SAs

About this task

You can set the maximum number of half-open IKE SAs and the maximum number of established IKE SAs.

- The supported maximum number of half-open IKE SAs depends on the device's processing capability. Adjust the maximum number of half-open IKE SAs to make full use of the device's processing capability without affecting the IKE SA negotiation efficiency.
- The supported maximum number of established IKE SAs depends on the device's memory space. Adjust the maximum number of established IKE SAs to make full use of the device's memory space without affecting other applications in the system.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the maximum number of half-open IKE SAs and the maximum number of established IKE SAs.

   ike limit { **max-negotiating-sa** *negotiation-limit* | **max-sa** *sa-limit* }

By default, the maximum number of half-open IKE SAs and IPsec SAs is 200, and there is no limit to the maximum number of established IKE SAs.

Configuring IKE address pools

About this task

To perform centralized management on remote users, an IPsec gateway can use an address pool to assign private IP addresses to remote users.

You must use an IKE address pool together with AAA authorization by specifying the IKE address pool as an AAA authorization attribute. For more information about AAA authorization, see "Configuring AAA."

Procedure

1. Enter system view.

   `system-view`

2. Configure an IKE IPv4 address pool.

   `ike address-group` *`group-name start-ipv4-address end-ipv4-address`* `[` *`mask`* `|` *`mask-length`* `]`

3. Configure an IKE IPv6 address pool.

   `ike ipv6-address-group` *`group-name`* **`prefix`** *`prefix/prefix-len`* **`assign-len`** *`assign-len`*

Configuring IKE negotiation compatibility

About this task

IKE negotiation between two peers using the SM4-CBC encryption algorithm will fail if the peers use different SM4-CBC key lengths. You can enable SM4-CBC key length compatibility on the device, so the device can successfully negotiate with a remote peer that uses a different SM4-CBC key length.

IKE peers running different software versions might have the GM main mode compatibility issue (signature verification failure) during IKE negotiation. If the device encounters this issue with its peer, you can enable the GM mode compatibility on the device. Do not enable GM mode compatibility on the device if the device does not have the compatibility issue with its peers.

Procedure

1. Enter system view.

   `system-view`

2. Enable SM4-CBC key length compatibility.

   `ike compatible-sm4 enable`

   By default, SM4-CBC key length compatibility is disabled.

3. Enable GM main mode compatibility.

   `ike compatible-gm-main enable`

   By default, the GM main mode compatibility is disabled.

Configuring GD-quantum encryption

About this task

After enabling GD-quantum encryption for IKE, the device can use the symmetric key provided by the GD-quantum server to encrypt the IPsec protected data, which further enhance the security of IPsec.

The device obtains keys from the GD-quantum server in the following steps:

1. **Connecting the GD-quantum server**—After all commands in IKE GD-quantum view are configured, the device establishes a connection with the specified GD-quantum server.
2. **Logging in to the GD-quantum server**—After the connection is established, the device sends a login request to the GD-quantum server, carrying the GD-quantum access ID and the GD-quantum authentication key. Only devices with verified GD-quantum access IDs and GD-quantum authentication keys can successfully log in to the GD-quantum server.
3. **Obtaining the GD-quantum keys**—After successful login and the IKE phase 1 negotiation, the device obtains the encrypted GD-quantum keys from the GD-quantum server, decrypt them with the GD-quantum decryption keys configured on the device, and finally obtain the GD-quantum keys for IPsec.

When enabling GD-quantum encryption for IKE, the device enters IKE GD-quantum view. Configure the IP address and port number of the GD-quantum server, the GD-quantum access IDs and GD-quantum authentication keys, and the GD-quantum decryption keys in this view.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480	Yes
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Prerequisites

Before configuring the GD-quantum encryption feature, contact the administrator of the GD-quantum server to obtain the GD-quantum access ID, the GD-quantum authentication key, and the GD-quantum decryption key.

Procedure

1. Enter system view.

   `system-view`

2. Enable GD-quantum encryption for IKE and enter IKE GD-quantum view.

   `ike gd-quantum`

   By default, GD-quantum encryption for IKE is disabled.

3. Specify the IP address and port number of the GD-quantum server.

   server-address *ip-address* [**port** *port-number*]

   By default, no IP address or port number of the GD-quantum server is specified.

4. Specify a GD-quantum access ID.

   app-dev-info *app-dev-info*

   By default, no GD-quantum access ID is specified.

5. Configure a GD-quantum authentication key.

   auth-key { **cipher** | **simple** } *key-value*

By default, no GD-quantum authentication key is configured.

6. Configure a GD-quantum decryption key.

   decrypt-quantum-key { **cipher** | **simple** } *key-value*

   By default, no GD-quantum decryption key is configured.

Configuring SNMP notifications and logging for IKE

Configuring SNMP notifications for IKE

About this task

After you enable SNMP notifications for IKE, the IKE module notifies the NMS of important module events. The notifications are sent to the device's SNMP module. For the notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP notifications, see *Network Management and Monitoring Configuration Guide*.

To generate and output SNMP notifications for a specific IKE failure or event type, perform the following tasks:

1. Enable SNMP notifications for IKE globally.
2. Enable SNMP notifications for the failure or event type.

Procedure

1. Enter system view

   system-view

2. Enable SNMP notifications for IKE globally.

   snmp-agent trap enable ike global

   By default, SNMP notifications for IKE are disabled.

3. Enable SNMP notifications for the specified failure or event types.

   snmp-agent trap enable ike [**attr-not-support** | **auth-failure** | **cert-type-unsupport** | **cert-unavailable** | **decrypt-failure** | **encrypt-failure** | **invalid-cert-auth** | **invalid-cookie** | **invalid-id** | **invalid-proposal** | **invalid-protocol** | **invalid-sign** | **no-sa-failure** | **proposal-add** | **proposal–delete** | **tunnel-start** | **tunnel-stop** | **unsupport-exch-type**] *

   By default, SNMP notifications for all failure and event types are disabled.

Enabling logging for IKE negotiation

About this task

This feature enables the device to output logs for the IKE negotiation process.

Procedure

1. Enter system view.

   system-view

2. Enable logging for IKE negotiation.

   ike logging negotiation enable

   By default, logging for IKE negotiation is enabled.

Display and maintenance commands for IKE

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display configuration information about all IKE proposals.	**display ike proposal**
Display information about the current IKE SAs.	**display ike sa** [**verbose** [**connection-id** *connection-id* | **remote-address** [**ipv6**] *remote-address* [**vpn-instance** *vpn-instance-name*]]]
Display IKE statistics.	**display ike statistics**
Delete IKE SAs.	**reset ike sa** [**connection-id** *connection-id*]
Clear IKE MIB statistics.	**reset ike statistics**

Troubleshooting IKE

IKE negotiation failed because no matching IKE proposals were found

Symptom

1. The IKE SA is in Unknown state.

```
<Sysname> display ike sa
    Connection-ID   Remote                Flag         DOI
------------------------------------------------------------------
    1               192.168.222.5/500     Unknown      IPsec
Flags:
RD--READY RL--REPLACED FD-FADING RK-REKEY
```

2. When IKE event debugging and packet debugging are enabled, the following messages appear:

   IKE event debugging message:

```
The attributes are unacceptable.
```

   IKE packet debugging message:

```
Construct notification packet: NO_PROPOSAL_CHOSEN.
```

Analysis

Certain IKE proposal settings are incorrect.

Solution

1. Examine the IKE proposal configuration to see whether the two ends have matching IKE proposals.
2. Modify the IKE proposal configuration to make sure the two ends have matching IKE proposals.
3. If the problem persists, contact NSFOCUS Support.

IKE negotiation failed because no IKE proposals or IKE keychains are specified correctly

Symptom

1. The IKE SA is in Unknown state.

```
<Sysname> display ike sa
    Connection-ID   Remote               Flag         DOI
------------------------------------------------------------------
    1               192.168.222.5/500     Unknown      IPsec
Flags:
RD--READY RL--REPLACED FD-FADING RK-REKEY
```

2. The following IKE event debugging or packet debugging message appeared:

   IKE event debugging message:

```
Notification PAYLOAD_MALFORMED is received.
```

   IKE packet debugging message:

```
Construct notification packet: PAYLOAD_MALFORMED.
```

Analysis

- If the following debugging information appeared, the matched IKE profile is not using the matched IKE proposal:

```
Failed to find proposal 1 in profile profile1.
```

- If the following debugging information appeared, the matched IKE profile is not using the matched IKE keychain:

```
Failed to find keychain keychain1 in profile profile1.
```

Solution

1. Verify that the matched IKE proposal (IKE proposal 1 in this debugging message example) is specified for the IKE profile (IKE profile 1 in the example).
2. Verify that the matched IKE keychain (IKE keychain 1 in this debugging message example) is specified for the IKE profile (IKE profile 1 in the example).
3. If the problem persists, contact NSFOCUS Support.

IPsec SA negotiation failed because no matching IPsec transform sets were found

Symptom

1. The **display ike sa** command shows that the IKE SA negotiation succeeded and the IKE SA is in RD state, but the **display ipsec sa** command shows that the expected IPsec SA has not been negotiated yet.
2. The following IKE debugging message appeared:

```
The attributes are unacceptable.
```

   Or:

```
Construct notification packet: NO_PROPOSAL_CHOSEN.
```

Analysis

Certain IPsec policy settings are incorrect.

Solution

1. Examine the IPsec configuration to see whether the two ends have matching IPsec transform sets.
2. Modify the IPsec configuration to make sure the two ends have matching IPsec transform sets.
3. If the problem persists, contact NSFOCUS Support.

IPsec SA negotiation failed due to invalid identity information

Symptom

1. The **`display ike sa`** command shows that the IKE SA negotiation succeeded and the IKE SA is in RD state, but the **`display ipsec sa`** command shows that the expected IPsec SA has not been negotiated yet.
2. The following IKE debugging message appeared:

   ```
   Notification INVALID_ID_INFORMATION is received.
   ```

   Or:

   ```
   Failed to get IPsec policy when renegotiating IPsec SA. Delete IPsec SA.
   Construct notification packet: INVALID_ID_INFORMATION.
   ```

Analysis

Certain IPsec policy settings of the responder are incorrect. Verify the settings as follows:

1. Verify that matching IKE profiles were found in IKE negotiation phase 1.

   If no matching IKE profiles were found, the IPsec policy must not use an IKE profile, so the global IKE settings can be used for IKE negotiation. If no matching IKE profiles were found and the IPsec policy is using an IKE profile, the IPsec SA negotiation fails.

   # Identify whether matching IKE profiles were found in IKE negotiation phase 1. The following output shows that no matching IKE profile was found:

   ```
   <Sysname> display ike sa verbose
      -----------------------------------------------
      Connection ID: 3
      Outside VPN:
      Inside VPN:
      Profile:
      Transmitting entity: Responder
      Initiator cookie: 1bcf453f0a217259
      Responder cookie: 5e32a74dfa66a0a4
      -----------------------------------------------
      Local IP/port: 192.168.222.5/500
      Local ID type: IPV4_ADDR
      Local ID: 192.168.222.5

      Remote IP/port: 192.168.222.71/500
      Remote ID type: IPV4_ADDR
      Remote ID: 192.168.222.71

      Authentication-method: PRE-SHARED-KEY
      Authentication-algorithm: MD5
      Encryption-algorithm: 3DES-CBC
   ```

```
   Life duration(sec): 86400
   Remaining key duration(sec): 85847
   Exchange-mode: Main
   Diffie-Hellman group: Group 1
   NAT traversal: Not detected
   Vendor ID index: 0xa1d
   Vendor ID sequence number: 0x0
```

# Identify whether the IPsec policy is using an IKE profile. The following output shows that no IKE profile is used by the IPsec policy.

```
[Sysname] display ipsec policy
-------------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet1/0/1
-------------------------------------------

  -----------------------------
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
  Description:
  Security data flow: 3000
  Selector mode: aggregation
  Local address: 192.168.222.5
  Remote address: 192.168.222.71
  Transform set:  transform1
  IKE profile: profile1
  smart-link policy:
  SA trigger mode: Auto
  SA duration(time based): 3600 seconds
  SA duration(traffic based): 1843200 kilobytes
  SA soft-duration buffer(time based): 1000 seconds
  SA soft-duration buffer(traffic based): 43200 kilobytes
  SA idle time: 100 seconds
```

2. Verify that the ACL specified for the IPsec policy is correctly configured. If the flow range defined by the responder's ACL is smaller than that defined by the initiator's ACL, IPsec proposal matching will fail.

   For example, if the initiator's ACL defines a flow from one network segment to another but the responder's ACL defines a flow from one host to another host, IPsec proposal matching will fail.

   # On the initiator:

```
[Sysname] display acl 3000
Advanced IPv4 ACL 3000, 1 rule,
ACL's step is 5
 rule 0 permit ip source 192.168.222.0 0.0.0.255 destination 192.168.222.0 0.0.0.255
```

   # On the responder:

```
[Sysname] display acl 3000
Advanced IPv4 ACL 3000, 1 rule,
ACL's step is 5
 rule 0 permit ip source 192.168.222.71 0 destination 192.168.222.5 0
```

3. Verify that the IPsec policy has a remote address and an IPsec transform set configured and that the IPsec transform set has all necessary settings configured.

   If, for example, the IPsec policy has no remote address configured, the IPsec SA negotiation will fail:

```
[Sysname] display ipsec policy
-------------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet1/0/1
-------------------------------------------

  -----------------------------
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
  Security data flow: 3000
  Selector mode: aggregation
  Local address: 192.168.222.5
  Remote address:
  Transform set:  transform1
  IKE profile: profile1
  smart-link policy:
  SA trigger mode: Auto
  SA duration(time based): 3600 seconds
  SA duration(traffic based): 1843200 kilobytes
  SA soft-duration buffer(time based): 1000 seconds
  SA soft-duration buffer(traffic based): 43200 kilobytes
  SA idle time: 100 seconds
```

Solution

1. If the IPsec policy specifies an IKE profile but no matching IKE profiles was found in IKE negotiation, perform one of the following tasks on the responder:
   - Remove the specified IKE profile from the IPsec policy.
   - Modify the specified IKE profile to match the IKE profile of the initiator.
2. If the flow range defined by the responder's ACL is smaller than that defined by the initiator's ACL, modify the responder's ACL so the ACL defines a flow range equal to or greater than that of the initiator's ACL.

   For example:

```
[Sysname] display acl 3000
Advanced IPv4 ACL 3000, 2 rules,
ACL's step is 5
 rule 0 permit ip source 192.168.222.0 0.0.0.255 destination 192.168.222.0 0.0.0.255
```

3. Configure the missing settings (for example, the remote address).
4. If the problem persists, contact NSFOCUS Support.

Configuring IKEv2

About IKEv2

Internet Key Exchange version 2 (IKEv2) is an enhanced version of IKEv1. The same as IKEv1, IKEv2 has a set of self-protection mechanisms and can be used on insecure networks for reliable identity authentication, key distribution, and IPsec SA negotiation. IKEv2 provides stronger protection against attacks and higher key exchange ability and needs fewer message exchanges than IKEv1.

IKEv2 negotiation process

Compared with IKEv1, IKEv2 simplifies the negotiation process and is much more efficient.

IKEv2 defines three types of exchanges: initial exchanges, CREATE_CHILD_SA exchange, and INFORMATIONAL exchange.

As shown in Figure 38, IKEv2 uses two exchanges during the initial exchange process: IKE_SA_INIT and IKE_AUTH, each with two messages.

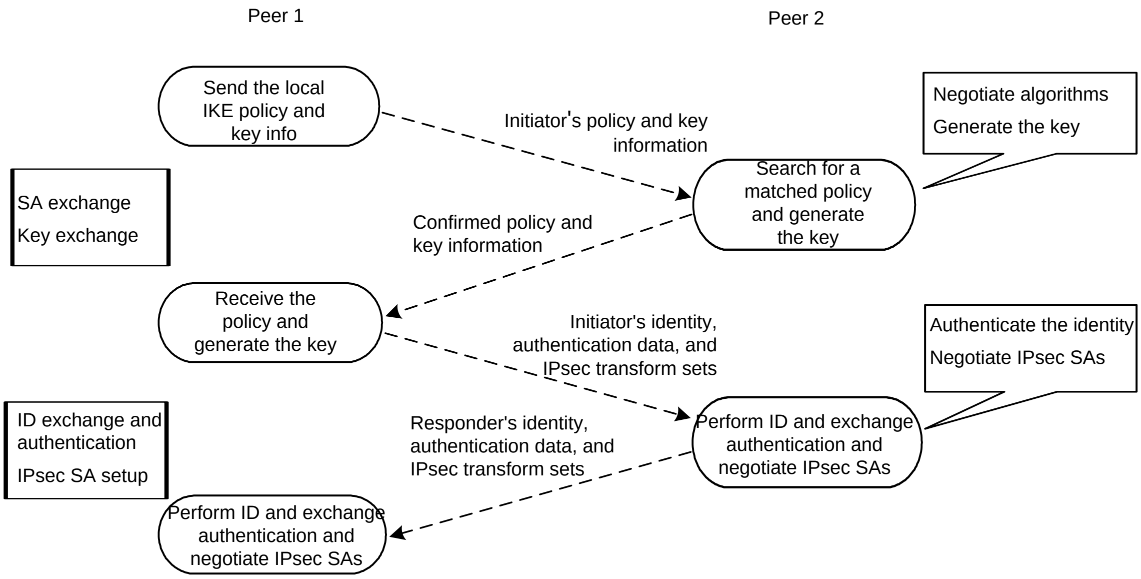
- **IKE_SA_INIT exchange**—Negotiates IKE SA parameters and exchanges keys.
- **IKE_AUTH exchange**—Authenticates the identity of the peer and establishes IPsec SAs.

After the four-message initial exchanges, IKEv2 sets up one IKE SA and one pair of IPsec SAs. For IKEv1 to set up one IKE SA and one pair of IPsec SAs, it must go through two phases that use a minimum of six messages.

To set up one more pair of IPsec SAs within the IKE SA, IKEv2 goes on to perform an additional two-message exchange—the CREATE_CHILD_SA exchange. One CREATE_CHILD_SA exchange creates one pair of IPsec SAs. IKEv2 also uses the CREATE_CHILD_SA exchange to rekey IKE SAs and Child SAs.

IKEv2 uses the INFORMATIONAL exchange to convey control messages about errors and notifications.

Figure 38 IKEv2 Initial exchange process



Peer 1
Peer 2
Send the local IKE policy and key info
Initiator's policy and key information
Negotiate algorithms
Generate the key
Search for a matched policy and generate the key
SA exchange
Key exchange
Confirmed policy and key information
Receive the policy and generate the key
Initiator's identity, authentication data, and IPsec transform sets
Authenticate the identity
Negotiate IPsec SAs
ID exchange and authentication
IPsec SA setup
Responder's identity, authentication data, and IPsec transform sets
Perform ID and exchange authentication and negotiate IPsec SAs
Perform ID and exchange authentication and negotiate IPsec SAs

New features in IKEv2

DH guessing

In the IKE_SA_INIT exchange, the initiator guesses the DH group that the responder is most likely to use and sends it in an IKE_SA_INIT request message. If the initiator's guess is correct, the responder responds with an IKE_SA_INIT response message and the IKE_SA_INIT exchange is finished. If the guess is wrong, the responder responds with an INVALID_KE_PAYLOAD message that contains the DH group that it wants to use. The initiator then uses the DH group selected by the responder to reinitiate the IKE_SA_INIT exchange. The DH guessing mechanism allows for more flexible DH group configuration and enables the initiator to adapt to different responders.

Cookie challenging

Messages for the IKE_SA_INIT exchange are in plain text. An IKEv1 responder cannot confirm the validity of the initiators and must maintain half-open IKE SAs, which makes the responder susceptible to DoS attacks. An attacker can send a large number of IKE_SA_INIT requests with forged source IP addresses to the responder, exhausting the responder's system resources.

IKEv2 introduces the cookie challenging mechanism to prevent such DoS attacks. When an IKEv2 responder maintains a threshold number of half-open IKE SAs, it starts the cookie challenging mechanism. The responder generates a cookie and includes it in the response sent to the initiator. If the initiator initiates a new IKE_SA_INIT request that carries the correct cookie, the responder considers the initiator valid and proceeds with the negotiation. If the carried cookie is incorrect, the responder terminates the negotiation.

The cookie challenging mechanism automatically stops working when the number of half-open IKE SAs drops below the threshold.

IKEv2 SA rekeying

For security purposes, both IKE SAs and IPsec SAs have a lifetime and must be rekeyed when the lifetime expires. An IKEv1 SA lifetime is negotiated. An IKEv2 SA lifetime, in contrast, is configured. If two peers are configured with different lifetimes, the peer with the shorter lifetime always initiates the SA rekeying. This mechanism reduces the possibility that two peers will simultaneously initiate a rekeying. Simultaneous rekeying results in redundant SAs and SA status inconsistency on the two peers.

IKEv2 message retransmission

Unlike IKEv1 messages, IKEv2 messages appear in request/response pairs. IKEv2 uses the Message ID field in the message header to identify the request/response pair. If an initiator sends a request but receives no response with the same Message ID value within a specific period of time, the initiator retransmits the request.

It is always the IKEv2 initiator that initiates the retransmission, and the retransmitted message must use the same Message ID value.

Protocols and standards

- RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*
- RFC 4306, *Internet Key Exchange (IKEv2) Protocol*
- RFC 4718, *IKEv2 Clarifications and Implementation Guidelines*
- RFC 2412, *The OAKLEY Key Determination Protocol*
- RFC 5996, *Internet Key Exchange Protocol Version 2 (IKEv2)*

IKEv2 tasks at a glance

To configure IKEv2, perform the following tasks:

1. Configuring an IKEv2 profile
   a. Creating an IKEv2 profile
   b. Specifying the local and remote identity authentication methods
   c. Configuring the IKEv2 keychain or PKI domain
   d. Configuring the local ID for the IKEv2 profile
   e. Configuring peer IDs for the IKEv2 profile
   f. Specifying a VPN instance for the IKEv2 profile
   g. Specifying an inside VPN instance for the IKEv2 profile
   h. Configuring optional features for the IKEv2 profile
2. Configuring an IKEv2 policy
3. Configuring an IKEv2 proposal

   If you specify an IKEv2 proposal in an IKEv2 policy, you must configure the IKEv2 proposal.
4. Configuring an IKEv2 keychain

   This task is required when either end or both ends use the preshared key authentication method.
5. (Optional.) Enabling the cookie challenging feature

   The cookie challenging feature takes effect only on IKEv2 responders.
6. (Optional.) Configuring the IKEv2 DPD feature
7. (Optional.) Configuring the IKEv2 NAT keepalive feature
8. (Optional.) Configuring IKEv2 address pools

Prerequisites for IKEv2 configuration

Determine the following parameters prior to IKEv2 configuration:

- The strength of the algorithms for IKEv2 negotiation, including the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups. Different algorithms provide different levels of protection. A stronger algorithm means better resistance to decryption of protected data but requires more resources. Typically, the longer the key, the stronger the algorithm.
- The local and remote identity authentication methods.
  - To use the preshared key authentication method, you must determine the preshared key.
  - To use the RSA digital signature authentication method, you must determine the PKI domain for the local end to use. For information about PKI, see "Configuring PKI."

Configuring an IKEv2 profile

Creating an IKEv2 profile

About this task

An IKEv2 profile is intended to provide a set of parameters for IKEv2 negotiation.

Procedure

1. Enter system view.

   ```
   system-view
   ```
2. Create an IKEv2 profile and enter its view.

   ```
   ikev2 profile profile-name
   ```

Specifying the local and remote identity authentication methods

Restrictions and guidelines

The local and remote identity authentication methods must both be specified and they can be different. You can specify only one local identity authentication method and multiple remote identity authentication methods.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Specify the local and remote identity authentication methods.

   authentication-method { **local** | **remote** } { **dsa-signature** | **ecdsa-signature** | **pre-share** | **rsa-signature** }

   By default, no local or remote identity authentication method is configured.

Configuring the IKEv2 keychain or PKI domain

Restrictions and guidelines

Configure the IKEv2 keychain or PKI domain for the IKEv2 profile to use. To use digital signature authentication, configure a PKI domain. To use preshared key authentication, configure an IKEv2 keychain.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Specify the keychain for preshared key authentication or the PKI domain used to request a certificate for digital signature authentication.

   - Specify the keychain.

     keychain *keychain-name*

   - Specify the PKI domain.

     certificate domain *domain-name* [**sign** | **verify**]

   By default, no IKEv2 keychain or PKI domain is specified for an IKEv2 profile.

Configuring the local ID for the IKEv2 profile

Restrictions and guidelines

For digital signature authentication, the device can use an ID of any type. If the local ID is an IP address that is different from the IP address in the local certificate, the device uses the FQDN as the local ID. The FQDN is the device name configured by using the **sysname** command.

For preshared key authentication, the device can use an ID of any type other than the DN.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Configure the local ID.

   identity local { **address** { *ipv4-address* | **ipv6** *ipv6-address* } | **dn** | **email** *email-string* | **fqdn** *fqdn-name* | **key-id** *key-id-string* }

   By default, no local ID is configured, and the device uses the IP address of the interface where the IPsec policy applies as the local ID.

Configuring peer IDs for the IKEv2 profile

About this task

Perform this task to configure the peer ID for IKEv2 profile matching. When the device needs to select an IKEv2 profile for IKEv2 negotiation with a peer, it compares the received peer ID with the peer IDs of its local IKE profiles. If a match is found, it uses the IKEv2 profile with the matching peer ID for negotiation. IKEv2 profiles will be compared in descending order of their priorities.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Configure a peer ID.

   match remote { **certificate** *policy-name* | **identity** { **address** { { *ipv4-address* [*mask* | *mask-length*] | **range** *low-ipv4-address high-ipv4-address* } | **ipv6** { *ipv6-address* [*prefix-length*] | **range** *low-ipv6-address high-ipv6-address* } } | **fqdn** *fqdn-name* | **email** *email-string* | **key-id** *key-id-string* } }

   You must configure a minimum of one peer ID on each of the two peers.

Specifying a VPN instance for the IKEv2 profile

About this task

After you specify a VPN instance for an IKEv2 profile, the IKEv2 profile can be used for IKEv2 negotiation only on the interfaces that belong to the VPN instance.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Specify a VPN instance for the IKEv2 profile.

   match vrf { **name** *vrf-name* | **any** }

   By default, an IKEv2 profile belongs to the public network.

Specifying an inside VPN instance for the IKEv2 profile

About this task

The inside VPN instance determines where the device should forward received IPsec packets after it de-encapsulates the packets. If you specify an inside VPN instance, the device looks for a route in the specified VPN instance to forward the packets. If you do not specify an inside VPN instance, the internal and external networks are in the same VPN instance. The device looks for a route in this VPN instance to forward the packets.

Restrictions and guidelines

The inside VPN instance specified in an IKEv2 profile takes effect only on IPsec policies that use the IKEv2 profile. It does not take effect on IPsec profiles that use the IKEv2 profile.

Procedure

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Specify an inside VPN instance.

   inside-vrf *vrf-name*

   By default, no inside VPN instance is specified for an IKEv2 profile. The internal and external networks are in the same VPN instance. The device forwards protected data to this VPN instance.

Configuring optional features for the IKEv2 profile

1. Enter system view.

   system-view

2. Enter IKEv2 profile view.

   ikev2 profile *profile-name*

3. Configure optional features as needed.

   - Configure IKEv2 DPD.

     dpd interval *interval* [**retry** *seconds*] { **on-demand** | **periodic** }

     By default, IKEv2 DPD is not configured for an IKEv2 profile and an IKEv2 profile uses the DPD settings configured in system view. If IKEv2 DPD is not configured in system view either, the device does not perform dead IKEv2 peer detection.

   - Specify the local interface or IP address to which the IKEv2 profile can be applied.

     match local address { *interface-type interface-number* | *ipv4-address* | **ipv6** *ipv6-address* }

     By default, an IKEv2 profile can be applied to any local interface or local IP address.

     Use this command to specify which address or interface can use the IKEv2 profile for IKEv2 negotiation. Specify the local address configured in IPsec policy or IPsec policy template view (using the **local-address** command) for this command. If no local address is configured, specify the IP address of the interface that uses the IPsec policy.

   - Specify a priority for the IKEv2 profile.

     priority *priority*

     By default, the priority of an IKEv2 profile is 100.

When the device needs to select an IKEv2 profile for IKEv2 negotiation with a peer, it compares the received peer ID with the peer ID of its local IKEv2 profiles in descending order of their priorities

- Set the IKEv2 SA lifetime for the IKEv2 profile.

  sa duration *seconds*

  By default, the IKEv2 SA lifetime is 86400 seconds.

  The local and remote ends can use different IKEv2 SA lifetimes and they do not negotiate the lifetime. The end with a smaller SA lifetime will initiate an SA negotiation when the lifetime expires.

- Set the IKEv2 NAT keepalive interval.

  nat-keepalive *seconds*

  By default, the global IKEv2 NAT keepalive setting is used.

  Configure this command when the device is behind a NAT gateway. The device sends NAT keepalive packets regularly to its peer to prevent the NAT session from being aged because of no matching traffic.

- Enable the configuration exchange feature.

  config-exchange { **request** | **set** { **accept** | **send** } }

  By default, all configuration exchange options are disabled.

  This feature applies to scenarios where the headquarters and branches communicate through virtual tunnels. It enables exchanges of IP address request and set messages between the IPsec gateway at a branch and the IPsec gateway at the headquarters.

  Table 2 Parameter descriptions

Parameter	Description
request	Enables the IPsec gateway at a branch to submit IP address request messages to the IPsec gateway at the headquarters.
set accept	Enables the IPsec gateway at a branch to accept the IP addresses pushed by the IPsec gateway at the headquarters.
set send	Enables the IPsec gateway at the headquarters to push IP addresses to IPsec gateways at branches.

- Enable AAA authorization.

  aaa authorization domain *domain-name* **username** *user-name*

  By default, AAA authorization is disabled.

  The AAA authorization feature enables IKEv2 to request authorization attributes, such as the IKEv2 address pool, from AAA. IKEv2 uses the address pool to assign IP addresses to remote users. For more information about AAA authorization, see "Configuring AAA."

Configuring an IKEv2 policy

About this task

During the IKE_SA_INIT exchange, each end tries to find a matching IKEv2 policy, using the IP address of the local security gateway as the matching criterion.

- If IKEv2 policies are configured, IKEv2 searches for an IKEv2 policy that uses the IP address of the local security gateway. If no IKEv2 policy uses the IP address or the policy is using an incomplete proposal, the IKE_SA_INIT exchange fails.
- If no IKEv2 policy is configured, IKEv2 uses the system default IKEv2 policy **default**.

The device matches IKEv2 policies in the descending order of their priorities. To determine the priority of an IKEv2 policy:

1. First, the device examines the existence of the **match local address** command. An IKEv2 policy with the **match local address** command configured has a higher priority.
2. If a tie exists, the device compares the priority numbers. An IKEv2 policy with a smaller priority number has a higher priority.
3. If a tie still exists, the device prefers an IKEv2 policy configured earlier.

Procedure

1. Enter system view.

   system-view
2. Create an IKEv2 policy and enter its view.

   ikev2 policy *policy-name*

   By default, an IKEv2 policy named **default** exists.
3. Specify the local interface or address used for IKEv2 policy matching.

   match local address { *interface-type interface-number* | *ipv4-address* | **ipv6** *ipv6-address* }

   By default, no local interface or address is used for IKEv2 policy matching, and the policy matches any local interface or address.
4. Specify a VPN instance for IKEv2 policy matching.

   match vrf { **name** *vrf-name* | **any** }

   By default, no VPN instance is specified for IKEv2 policy matching. The IKEv2 policy matches all local addresses in the public network.
5. Specify an IKEv2 proposal for the IKEv2 policy.

   proposal *proposal-name*

   By default, no IKEv2 proposal is specified for an IKEv2 policy.
6. Specify a priority for the IKEv2 policy.

   priority *priority*

   By default, the priority of an IKEv2 policy is 100.

Configuring an IKEv2 proposal

About this task

An IKEv2 proposal contains security parameters used in IKE_SA_INIT exchanges, including the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups. An algorithm specified earlier has a higher priority.

Restrictions and guidelines

A complete IKEv2 proposal must have at least one set of security parameters, including one encryption algorithm, one integrity protection algorithm, one PRF algorithm, and one DH group.

You can specify multiple IKEv2 proposals for an IKEv2 policy. A proposal specified earlier has a higher priority.

Procedure

1. Enter system view.

   system-view
2. Create an IKEv2 proposal and enter its view.

   ikev2 proposal *proposal-name*

By default, an IKEv2 proposal named **default** exists.

The default proposal uses the following settings:

- Encryption algorithms AES-CBC-128 and 3DES.
- Integrity protection algorithms HMAC-SHA1 and HMAC-MD5.
- PRF algorithms HMAC-SHA1 and HMAC-MD5.
- DH groups 2 and 5.

3. Specify the encryption algorithms.

   encryption { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256** | **camellia-cbc-128** | **camellia-cbc-192** | **camellia-cbc-256** | **des-cbc** } *

   By default, an IKEv2 proposal does not have any encryption algorithms.
4. Specify the integrity protection algorithms.

   integrity { **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512** } *

   By default, an IKEv2 proposal does not have any integrity protection algorithms.
5. Specify the DH groups.

   dh { **group1** | **group14** | **group2** | **group24** | **group5** | **group19** | **group20** } *

   By default, an IKEv2 proposal does not have any DH groups.
6. Specify the PRF algorithms.

   prf { **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512** } *

   By default, an IKEv2 proposal uses the integrity protection algorithms as the PRF algorithms.

Configuring an IKEv2 keychain

About this task

An IKEv2 keychain specifies the preshared keys used for IKEv2 negotiation.

An IKEv2 keychain can have multiple IKEv2 peers. Each peer has a symmetric preshared key or an asymmetric preshared key pair, and information for identifying the peer (such as the peer's host name, IP address or address range, or ID).

An IKEv2 negotiation initiator uses the peer host name or IP address/address range as the matching criterion to search for a peer. A responder uses the peer host IP address/address range or ID as the matching criterion to search for a peer.

Procedure

1. Enter system view.

   system-view
2. Create an IKEv2 keychain and enter its view.

   ikev2 keychain *keychain-name*
3. Create an IKEv2 peer and enter its view.

   peer *name*
4. Configure a host name for the peer:

   hostname *name*

   By default, no host name is configured for an IKEv2 peer.
5. Configure a host IP address or address range for the peer:

address { *ipv4-address* [*mask* | *mask-length*] | **ipv6** *ipv6-address* [*prefix-length*] }

By default, no host IP address or address range is configured for an IKEv2 peer.

You must configure different host IP addresses/address ranges for different peers.

6. Configure an ID for the peer:

   identity { **address** { *ipv4-address* | **ipv6** { *ipv6-address* } } | **fqdn** *fqdn-name* | **email** *email-string* | **key-id** *key-id-string* }

   By default, no identity information is configured for an IKEv2 peer.

7. Configure a preshared key for the peer.

   pre-shared-key [**local** | **remote**] { **ciphertext** | **plaintext** } *string*

   By default, an IKEv2 peer does not have a preshared key.

Configure global IKEv2 parameters

Enabling the cookie challenging feature

About this task

Enable cookie challenging on responders to protect them against DoS attacks that use a large number of source IP addresses to forge IKE_SA_INIT requests.

Procedure

1. Enter system view.

   system-view

2. Enable IKEv2 cookie challenging.

   ikev2 cookie-challenge *number*

   By default, IKEv2 cookie challenging is disabled.

Configuring the IKEv2 DPD feature

About this task

IKEv2 DPD detects dead IKEv2 peers in periodic or on-demand mode.

- **Periodic IKEv2 DPD**—Verifies the liveness of an IKEv2 peer by sending DPD messages at regular intervals.
- **On-demand IKEv2 DPD**—Verifies the liveness of an IKEv2 peer by sending DPD messages before sending data.
  - Before the device sends data, it identifies the time interval for which the last IPsec packet has been received from the peer. If the time interval exceeds the DPD interval, it sends a DPD message to the peer to detect its liveliness.
  - If the device has no data to send, it never sends DPD messages.

Restrictions and guidelines

If you configure IKEv2 DPD in both IKEv2 profile view and system view, the IKEv2 DPD settings in IKEv2 profile view apply. If you do not configure IKEv2 DPD in IKEv2 profile view, the IKEv2 DPD settings in system view apply.

Procedure

1. Enter system view.

   system-view

2. Configure global IKEv2 DPD.

   ikev2 dpd interval *interval* [**retry** *seconds*] { **on-demand** | **periodic** }

   By default, global DPD is disabled.

Configuring the IKEv2 NAT keepalive feature

About this task

Configure this feature on the IKEv2 gateway behind the NAT device. The gateway then sends NAT keepalive packets regularly to its peer to keep the NAT session alive, so that the peer can access the device.

The NAT keepalive interval must be shorter than the NAT session lifetime.

This feature takes effect after the device detects the NAT device.

Procedure

1. Enter system view.

   system-view

2. Set the IKEv2 NAT keepalive interval.

   ikev2 nat-keepalive *seconds*

   By default, the IKEv2 NAT keepalive interval is 10 seconds.

Configuring IKEv2 address pools

About this task

To perform centralized management on remote users, an IPsec gateway can use an address pool to assign private IP addresses to remote users.

You must use an IKEv2 address pool together with AAA authorization by specifying the IKEv2 address pool as an AAA authorization attribute. For more information about AAA authorization, see "Configuring AAA."

Procedure

1. Enter system view.

   system-view

2. Configure an IKEv2 IPv4 address pool.

   ikev2 address-group *group-name start-ipv4-address end-ipv4-address* [*mask* | *mask-length*]

3. Configure an IKEv2 IPv6 address pool.

   ikev2 ipv6-address-group *group-name* **prefix** *prefix/prefix-len* **assign-len** *assign-len*

Display and maintenance commands for IKEv2

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the IKEv2 policy configuration.	**display ikev2 policy** [*policy-name* | **default**]

Task	Command
Display the IKEv2 profile configuration.	**display ikev2 profile** [*profile-name*]
Display the IKEv2 proposal configuration.	**display ikev2 proposal** [*name* | **default**]
Display the IKEv2 SA information.	**display ikev2 sa** [**count** | [{ **local** | **remote** } { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]] [**verbose** [**tunnel** *tunnel-id*]]]
Display IKEv2 statistics.	**display ikev2 statistics**
Delete IKEv2 SAs and the child SAs negotiated through the IKEv2 SAs.	**reset ikev2 sa** [[{ **local** | **remote** } { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]] | **tunnel** *tunnel-id*] [**fast**]
Clear IKEv2 statistics.	**reset ikev2 statistics**

Troubleshooting IKEv2

IKEv2 negotiation failed because no matching IKEv2 proposals were found

Symptom

The IKEv2 SA is in IN-NEGO status.

```
<Sysname> display ikev2 sa
Tunnel ID   Local                          Remote                         Status
  ---------------------------------------------------------------------------
  5          123.234.234.124/500         123.234.234.123/500         IN-NEGO
Status:
IN-NEGO: Negotiating, EST: Established, DEL:Deleting
```

Analysis

Certain IKEv2 proposal settings are incorrect.

Solution

1. Examine the IKEv2 proposal configuration to see whether the two ends have matching IKEv2 proposals.
2. Modify the IKEv2 proposal configuration to make sure the two ends have matching IKEv2 proposals.
3. If the problem persists, contact NSFOCUS Support.

IPsec SA negotiation failed because no matching IPsec transform sets were found

Symptom

The **`display ikev2 sa`** command shows that the IKEv2 SA negotiation succeeded and the IKEv2 SA is in EST status. The **`display ipsec sa`** command shows that the expected IPsec SAs have not been negotiated yet.

Analysis

Certain IPsec policy settings are incorrect.

Solution

1. Examine the IPsec configuration to see whether the two ends have matching IPsec transform sets.
2. Modify the IPsec configuration to make sure the two ends have matching IPsec transform sets.
3. If the problem persists, contact NSFOCUS Support.

IPsec tunnel establishment failed

Symptom

The ACLs and IKEv2 proposals are correctly configured on both ends. The two ends cannot establish an IPsec tunnel or cannot communicate through the established IPsec tunnel.

Analysis

The IKEv2 SA or IPsec SAs on either end are lost. The reason might be that the network is unstable and the device reboots.

Solution

1. Use the **`display ikev2 sa`** command to examine whether an IKEv2 SA exists on both ends. If the IKEv2 SA on one end is lost, delete the IKEv2 SA on the other end by using the **`reset ikev2 sa`** command and trigger new negotiation. If an IKEv2 SA exists on both ends, go to the next step.
2. Use the **`display ipsec sa`** command to examine whether IPsec SAs exist on both ends. If the IPsec SAs on one end are lost, delete the IPsec SAs on the other end by using the **`reset ipsec sa`** command and trigger new negotiation.
3. If the problem persists, contact NSFOCUS Support.

Contents

Configuring tunneling

This chapter describes tunnel interface configuration. For information about tunnel modes, see the subsequent chapters.

About tunneling

Tunneling encapsulates the packets of a network protocol within the packets of a second network protocol and transfers them over a virtual point-to-point connection. The virtual connection is called a tunnel. Packets are encapsulated at the tunnel source and de-encapsulated at the tunnel destination.

Supported tunneling technologies

Tunneling supports the following technologies:

- ADVPN tunneling. For more information, see "Configuring ADVPN."
- GRE tunneling. For more information, see "Configuring GRE."
- IPsec tunneling. For more information, see *Security Configuration Guide*.
- VXLAN tunneling. For more information, see *VXLAN Configuration Guide*.
- IPv6 over IPv4 tunneling, IPv4 over IPv4 tunneling, and IPv4/IPv6 over IPv6 tunneling.

Restrictions and guidelines: Tunnel interface configuration

Do not specify the same tunnel source and destination addresses for different tunnels on the same device.

If the length for the tunnel headers of a packet is too long, the device might fail to process the packet. As a best practice, the number of nested encapsulations for a packet cannot exceed 5.

On IPv4 and IPv6 VXLAN tunnel interfaces and VXLAN-DCI tunnel interfaces, IP addresses (assigned by **ip address** and **ipv6 address** commands) are meaningless. As a best practice, do not assign IP addresses to those types of tunnel interfaces.

Configuring a tunnel interface

About tunnel interface configuration

Configure a tunnel interface (Layer 3 virtual interface) at both ends of a tunnel. The devices use the tunnel interface to identify, process, and send packets for the tunnel.

Tunnel interface configuration tasks at a glance

To configure a tunnel interface, perform the following tasks:

1. Creating a tunnel interface
2. (Optional.) Configuring parameters for tunneled packets

3. (Optional.) Specifying the tunnel destination VPN instance
4. (Optional.) Restoring the default settings of a tunnel interface

Creating a tunnel interface

1. Enter system view.

   system-view

2. Create a tunnel interface, specify the tunnel mode, and enter tunnel interface view.

   interface tunnel *number* **mode** { **advpn** { **gre** | **udp** } [**ipv6**] | **ds-lite-aftr** | **gre** [**ipv6**] | **gre-p2mp** [**ipv6**] | **ipsec** [**ipv6**] | **ipv4-ipv4** | **ipv6** | **ipv6-ipv4** [**6rd** | **6to4** | **auto-tunnel** | **isatap**] | **vxlan** }

   For packet tunneling to succeed, the two ends of a tunnel must use the same tunnel mode.

   Support for the following keywords depends on the device model:

   - **advpn** { **gre** | **udp** } [**ipv6**].
   - **gre-p2mp** [**ipv6**].
   - **ipsec** [**ipv6**].
   - **ipv6-ipv4 6rd**.
   - **vxlan**.

   For more information, see the command reference.

3. Configure a source address or source interface for the tunnel.

   source { *ipv4-address* | *ipv6-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

4. Configure a destination address for the tunnel.

   destination { *ipv4-address* | *ipv6-address* | **dhcp-alloc** *interface-type interface-number* }

   By default, no destination address is configured for a tunnel.

   The tunnel destination address must be the IP address of the receiving interface on the tunnel peer. It is used as the destination IP address of tunneled packets.

5. (Optional.) Configure a description for the interface.

   description *text*

   By default, the description for a tunnel interface is **Tunnel** *number* **Interface**.

6. (Optional.) Set the MTU of the tunnel interface.

   mtu *size*

   The default settings are as follows:

   - If the tunnel interface has never been up, the MTU is 64000 bytes.
   - If the tunnel interface is up, its MTU is identical to the outgoing interface's MTU minus the length of the tunnel headers. The outgoing interface is automatically obtained through routing table lookup based on the tunnel destination address.

7. (Optional.) Set the expected bandwidth for the tunnel interface.

   bandwidth *bandwidth-value*

   The default expected bandwidth (in kbps) is the interface maximum rate divided by 1000.

The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

8. Bring up the tunnel interface.

   undo shutdown

   By default, a tunnel interface is not administratively down.

Configuring parameters for tunneled packets

1. Enter system view.

   system-view

2. Enter tunnel interface view.

   interface tunnel *number*

3. Set the ToS for tunneled packets.

   tunnel tos { **copy-inner-tos** | *tos-value* }

   The default settings are as follows:

   - For VXLAN tunneled packets, the ToS is 0.
   - For non-VXLAN tunneled packets, the ToS is the same as the ToS of the original packets.

   The **copy-inner-tos** keyword is supported only by VXLAN tunnels.

4. Set the TTL for tunneled packets.

   tunnel ttl *ttl-value*

   The default TTL for tunneled packets is 255.

Specifying the tunnel destination VPN instance

Restrictions and guidelines

For a tunnel interface to come up, the tunnel source and destination must belong to the same VPN instance. To specify a VPN instance for the tunnel source, use the **ip binding vpn-instance** command on the tunnel source interface. For more information about this command, see *VPN instance in VPN Command Reference*.

Procedure

1. Enter system view.

   system-view

2. Enter tunnel interface view.

   interface tunnel *number*

3. Specify the VPN instance to which the tunnel destination belongs.

   tunnel vpn-instance *vpn-instance-name*

   By default, the tunnel destination belongs to the public network.

Restoring the default settings of a tunnel interface

Restrictions and guidelines

△ **CAUTION:**

This operation might interrupt ongoing network services. Make sure you are fully aware of the impact of this operation when you perform it on a live network.

This operation might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **display this** command in interface view to identify these commands. Use their **undo** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.
   system-view
2. Enter tunnel interface view.
   interface tunnel *number*
3. Restore the default settings of the tunnel interface.
   default

Display and maintenance commands for tunnel interface configuration

Execute **display** commands in any view and **reset** commands in user view.

Task	Command	Remarks
Display information about tunnel interfaces.	**display interface** [**tunnel** [*number*]] [**brief** [**description** | **down**]]	N/A
Display IPv6 information on tunnel interfaces.	**display ipv6 interface** [**tunnel** [*number*]] [**brief**]	For more information about this command, see IPv6 basics in *Layer 3—IP Services Command Reference*.
Clear statistics on tunnel interfaces.	**reset counters interface** [**tunnel** [*number*]]	N/A
Clear IPv6 statistics on tunnel interfaces.	**reset ipv6 statistics** [**slot** *slot-number*]	For more information about this command, see IPv6 basics in *Layer 3—IP Services Command Reference*.

Troubleshooting tunnel interface configuration

Tunnel interface not up

Symptom

A tunnel interface configured with related parameters such as tunnel source address, tunnel destination address, and tunnel mode cannot come up.

Analysis

The physical interface of the tunnel does not come up, or the tunnel destination is unreachable.

Solution

1. To resolve the problem:

- Use the **`display interface`** or **`display ipv6 interface`** command to verify that the physical interface of the tunnel is up. If the physical interface is down, check the network connection.
- Use the **`display ipv6 routing-table`** or **`display ip routing-table`** command to verify that the tunnel destination is reachable. If the route is not available, configure a route to reach the tunnel destination.

2. If the problem persists, contact NSFOCUS Support.

IPv6 over IPv4 tunneling

About IPv6 over IPv4 tunneling

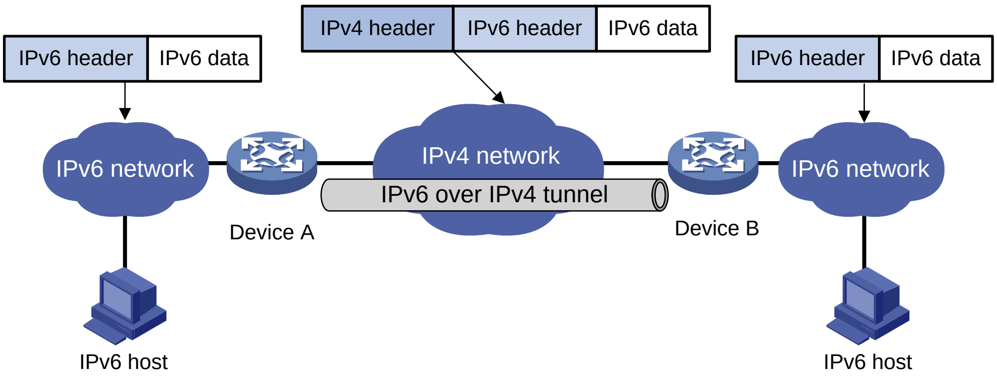
Implementation

IPv6 over IPv4 tunneling enables isolated IPv6 networks to communicate, as shown in Figure 1.

NOTE:

The devices at both ends of an IPv6 over IPv4 tunnel must support the IPv4/IPv6 dual stack.

Figure 1 IPv6 over IPv4 tunnel



The IPv6 over IPv4 tunnel processes packets by using the following steps:

1. A host in the IPv6 network sends an IPv6 packet to Device A at the tunnel source.
2. After Device A receives the IPv6 packet, it processes the packet as follows:
   a. Searches the routing table to identify the outgoing interface for the IPv6 packet.

      The outgoing interface is the tunnel interface, so Device A knows that the packet needs to be forwarded through the tunnel.

   b. Adds an IPv4 header to the IPv6 packet and forwards the packet through the physical interface of the tunnel.

      In the IPv4 header, the source IPv4 address is the IPv4 address of the tunnel source, and the destination IPv4 address is the IPv4 address of the tunnel destination.
3. Upon receiving the packet, Device B de-encapsulates the packet.
4. If the destination address of the IPv6 packet is itself, Device B forwards it to the upper-layer protocol. If it is not, Device B forwards it according to the routing table.

Tunnel modes

IPv6 over IPv4 tunnels include manually configured tunnels and automatic tunnels, depending on how the IPv4 address of the tunnel destination is obtained.

- **Manually configured tunnel**—The destination IPv4 address of the tunnel cannot be automatically obtained from the destination IPv6 address of an IPv6 packet at the tunnel source. It must be manually configured.

- **Automatic tunnel**—The destination IPv4 address of the tunnel can be automatically obtained from the destination IPv6 address (with an IPv4 address embedded) of an IPv6 packet at the tunnel source.

The source IPv4 addresses for all IPv6 over IPv4 tunnels are manually configured.

According to the way an IPv6 packet is encapsulated, IPv6 over IPv4 tunnels are divided into the modes shown in the following sections.

IPv6 over IPv4 manual tunneling

An IPv6 over IPv4 manual tunnel is a point-to-point link. To establish a manual tunnel, you must manually configure the source and destination addresses of the tunnel at both ends of the tunnel.

Manual tunneling provides the following solutions:

- Connects isolated IPv6 networks over an IPv4 network.
- Connects an IPv6 network and an IPv4/IPv6 dual-stack host over an IPv4 network.

Automatic IPv4-compatible IPv6 tunneling

An automatic IPv4-compatible IPv6 tunnel is a point-to-multipoint link. The ends of an automatic IPv4-compatible IPv6 tunnel are IPv4-compatible IPv6 addresses. The address format is 0:0:0:0:0:0:a.b.c.d/96, where a.b.c.d is an IPv4 address. The destination IPv4 address of an automatic IPv4-compatible IPv6 tunnel is embedded in the destination IPv4-compatible IPv6 address. This mechanism enables the device to automatically obtain the tunnel destination address.

Automatic IPv4-compatible IPv6 tunnels have limitations because IPv4-compatible IPv6 addresses must use globally unique IPv4 addresses.

6to4 tunneling

- Ordinary 6to4 tunneling

  A 6to4 tunnel is a point-to-multipoint automatic tunnel. It is used to connect multiple isolated IPv6 networks over an IPv4 network.

  The ends of a 6to4 tunnel are 6to4 addresses. The address format is 2002:abcd:efgh:subnet number::interface ID/48.

  - 2002 is the fixed IPv6 address prefix.
  - abcd:efgh represents a 32-bit globally unique IPv4 address in hexadecimal notation.

    For example, 1.1.1.1 can be represented by 0101:0101. The IPv4 address identifies a 6to4 network (an IPv6 network where all hosts use 6to4 addresses). The border router of a 6to4 network must have the IPv4 address abcd:efgh configured on the interface connected to the IPv4 network.
  - The subnet number identifies a subnet in the 6to4 network.
  - The subnet number::interface ID uniquely identifies a host in the 6to4 network.

  The destination IPv4 address of a 6to4 tunnel is embedded in the destination 6to4 address. This mechanism enables the device to automatically obtain the tunnel destination address.

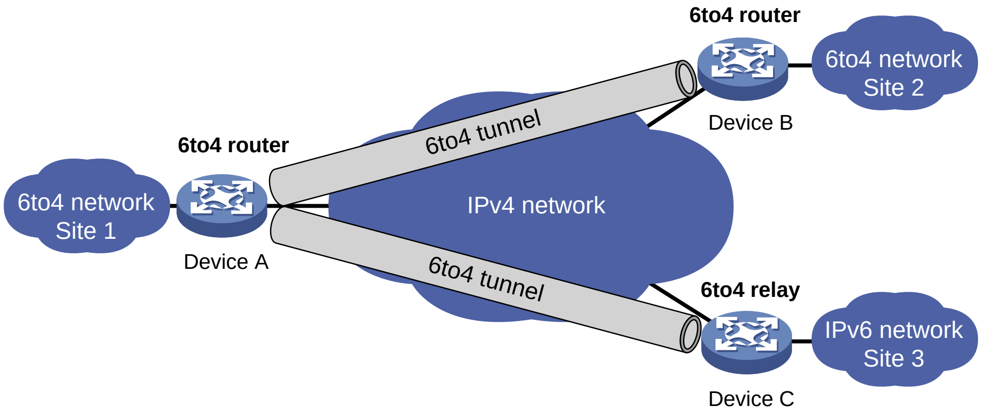
  6to4 tunneling uses an IPv4 address to identify a 6to4 network. This method overcomes the limitations of automatic IPv4-compatible IPv6 tunneling.

- 6to4 relay

  6to4 relay connects a 6to4 network and an IPv6 network that uses an IP prefix other than 2002::/16. A 6to4 relay router is a gateway that forwards packets from a 6to4 network to an IPv6 network.

  As shown in Figure 2, 6to4 network Site 1 communicates with IPv6 network Site 3 over a 6to4 tunnel. Configure a static route on the border router (Device A) in the 6to4 network. The next hop address must be the 6to4 address of the 6to4 relay router (Device C). Device A forwards all packets destined for the IPv6 network over the 6to4 tunnel, and Device C then forwards them to the IPv6 network.

Figure 2 Principle of 6to4 tunneling and 6to4 relay



ISATAP tunneling

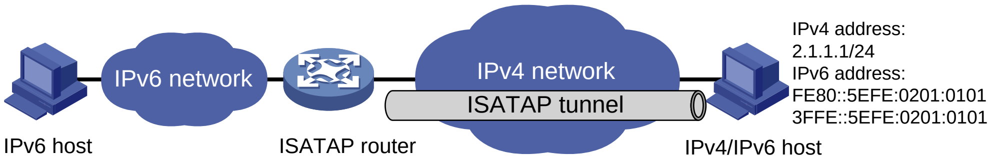
An ISATAP tunnel is a point-to-multipoint automatic tunnel. It provides a solution to connect an IPv6 host and an IPv6 network over an IPv4 network.

The destination address of an ISATAP tunnel is an ISATAP address. The address format is prefix:0:5EFE:abcd:efgh/64.

- The 64-bit prefix is a valid IPv6 unicast address prefix.
- The abcd:efgh/64 segments represent a 32-bit IPv4 address in hexadecimal notation, which identifies the tunnel destination but does not require global uniqueness.

ISATAP tunnels are mainly used for communication between IPv6 routers or between an IPv6 host and an IPv6 router over an IPv4 network.

Figure 3 Principle of ISATAP tunneling



6RD tunneling

- Ordinary 6RD tunneling

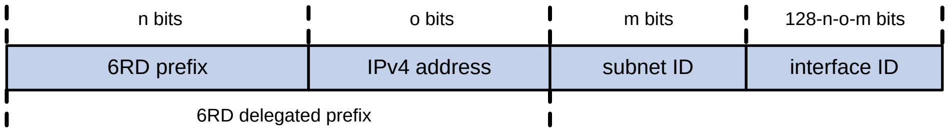
  A 6RD tunnel is a point-to-multipoint automatic tunnel. It is an extension of 6to4 tunneling. The IPv6 prefix of a 6RD network is assigned by the service provider and is not limited to the prefix 2002::/16. 6RD does not require embedding all 32 bits of an IPv4 address in a 6RD address.

  Figure 4 shows the format of 6RD address—an IPv6 address with a 6RD prefix and an embedded IPv4 address.

  - **6RD prefix**—IPv6 prefix assigned by the service provider to a 6RD network.
  - **IPv4 address**—All or part of the IPv4 tunnel source address in hexadecimal notation. For example, assume that the IPv4 tunnel source address is 1.2.3.4 and both the IPv4 prefix length and suffix length are specified as 8 bits. Then, the prefix of the IPv4 address 1.2.3.4 is 01, the suffix is 04, and the embedded IPv4 address is 0203.
  - **6RD delegated prefix**—The 6RD prefix and the embedded IPv4 address together form a 6RD delegated prefix. It uniquely identifies a 6RD network (an IPv6 network where all hosts use 6RD addresses).
  - **Subnet ID**—Identifies a subnet in the 6RD network.
  - **Interface ID**—The subnet ID and the interface ID together identify a host in the 6RD network.

The device automatically identifies the tunnel destination IPv4 address according to the IPv4 address embedded in the 6RD address, the 6RD prefix, and the IPv4 prefix and suffix.

Figure 4 6RD address format

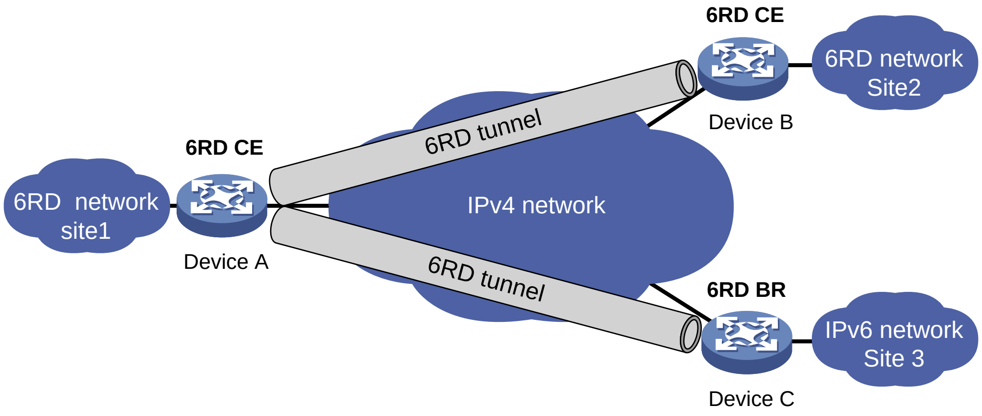


- 6RD relay

  6 RD relay connects a 6RD network and an IPv6 network that uses a prefix other than a 6RD delegated prefix. A 6RD border relay (BR) router is a gateway that forwards packets from a 6RD network to an IPv6 network. The border router connected to the 6RD network is called 6RD CE.

  As shown in Figure 5, for the 6RD network site 1 to communicate with the IPv6 network site 3 over a 6RD tunnel, you must configure 6RD relay. Configure a static route on the border router (Device A) in the 6RD network. The next hop address must be the 6RD address of the 6RD BR router (Device C). Device A forwards all packets destined for the IPv6 network over the 6RD tunnel, and Device C then forwards them to the IPv6 network.

Figure 5 Principle of 6RD tunneling and 6RD relay



IPv6 over IPv4 tunneling tasks at a glance

To configure IPv6 over IPv4 tunneling, perform the following tasks:

1. Configuring an IPv6 over IPv4 tunnel

   Choose one of the following tasks:

   - Configuring an IPv6 over IPv4
   - Configuring an automatic IPv4-compatible IPv6 tunnel
   - Configuring a 6to4 tunnel
   - Configuring an ISATAP tunnel
   - Configuring a 6RD tunnel
2. (Optional.) Enabling dropping IPv6 packets that use IPv4-compatible IPv6 addresses

Configuring an IPv6 over IPv4 manual tunnel

Restrictions and guidelines

When you perform tasks in this section, follow these restrictions and guidelines:

- The tunnel destination address specified on the local device must be identical with the tunnel source address specified on the tunnel peer device.
- Do not specify the same tunnel source and destination addresses for different tunnels in the same mode on a device.
- To ensure correct packet forwarding, identify whether the destination IPv6 network and the IPv6 address of the local tunnel interface are on the same subnet. If they are not, configure a route reaching the destination IPv6 network through the tunnel interface. You can configure the route by using one of the following methods:
  - Configure a static route, and specify the local tunnel interface as the egress interface or specify the IPv6 address of the peer tunnel interface as the next hop.
  - Enable a dynamic routing protocol on both the local and remote tunnel interfaces.

  For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter IPv6 over IPv4 manual tunnel interface view.

   interface tunnel *number* [**mode ipv6-ipv4**]

3. Specify an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."

4. Configure a source address or source interface for the tunnel.

   source { *ipv4-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

5. Configure a destination address for the tunnel.

   destination *ipv4-address*

   By default, no destination address is configured for a tunnel.

   The tunnel destination address must be the IP address of the receiving interface on the tunnel peer. It is used as the destination IP address of tunneled packets.

6. (Optional.) Set the DF bit for tunneled packets.

   tunnel dfbit enable

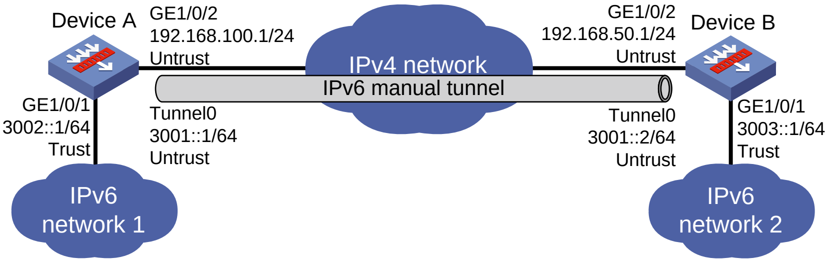
   By default, the DF bit is not set for tunneled packets.

Example: Configuring an IPv6 over IPv4 manual tunnel

Network configuration

As shown in Figure 6, configure an IPv6 over IPv4 tunnel between Device A and Device B so the two IPv6 networks can reach each other over the IPv4 network. Because the tunnel destination IPv4 address cannot be automatically obtained from the destination IPv6 addresses, configure an IPv6 over IPv4 manual tunnel.

Figure 6 Network diagram



Procedure

1. Configure Device A:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 3002:1/64
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 192.168.100.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/2] quit
```

# Create IPv6 over IPv4 manual tunnel interface **Tunnel 0**.

```
[DeviceA] interface tunnel 0 mode ipv6-ipv4
[DeviceA-Tunnel0] ipv6 address 3001::1/64
[DeviceA-Tunnel0] source gigabitethernet 1/0/2
[DeviceA-Tunnel0] destination 192.168.50.1
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 192.168.100.2.

```
[DeviceA] ip route-static 192.168.50.1 24 192.168.100.2
[DeviceA] ipv6 route-static 3003:: 64 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 192.168.100.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 192.168.50.1
```

```
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 192.168.50.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 192.168.100.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from IPv6 network 1 to IPv6 network 2.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-ip-subnet 3002:: 64
[DeviceA-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 3003:: 64
[DeviceA-security-policy-ipv6-3-trust-untrust] action pass
[DeviceA-security-policy-ipv6-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from IPv6 network 2 to IPv6 network 1.

```
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-ip-subnet 3003:: 64
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-ip-subnet 3002:: 64
[DeviceA-security-policy-ipv6-4-untrust-trust] action pass
[DeviceA-security-policy-ipv6-4-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 3003::1 64
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 192.168.50.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create IPv6 over IPv4 manual tunnel interface **Tunnel 0**.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4
[DeviceB-Tunnel0] ipv6 address 3001::2/64
[DeviceB-Tunnel0] source gigabitethernet 1/0/2
[DeviceB-Tunnel0] destination 192.168.100.1
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 192.168.50.2.

```
[DeviceB] ip route-static 192.168.100.1 24 192.168.50.2
[DeviceB] ipv6 route-static 3002:: 64 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 192.168.50.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 192.168.100.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 192.168.100.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 192.168.50.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from IPv6 network 2 to IPv6 network 1.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-ip-subnet 3003:: 64
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 3002:: 64
[DeviceB-security-policy-ipv6-3-trust-untrust] action pass
[DeviceB-security-policy-ipv6-3-trust-untrust] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from IPv6 network 1 to IPv6 network 2.

```
[DeviceB-security-policy-ipv6] rule name untrust-trust
[DeviceB-security-policy-ipv6-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ipv6-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ipv6-4-untrust-trust] source-ip-subnet 3002:: 64
[DeviceB-security-policy-ipv6-4-untrust-trust] destination-ip-subnet 3003:: 64
```

```
[DeviceB-security-policy-ipv6-4-untrust-trust] action pass
[DeviceB-security-policy-ipv6-4-untrust-trust] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Use the **`display ipv6 interface`** command to display tunnel interface status on Device A and Device B. Verify that interface tunnel 0 is up. (Details not shown.)

# Verify that Device B and Device A can ping the IPv6 address of GigabitEthernet 1/0/1 of each other. This example uses Device A.

```
[DeviceA] ping ipv6 3003::1
Ping6(56 data bytes) 3001::1 --> 3003::1, press CTRL C to break
56 bytes from 3003::1, icmp_seq=0 hlim=64 time=45.000 ms
56 bytes from 3003::1, icmp_seq=1 hlim=64 time=10.000 ms
56 bytes from 3003::1, icmp_seq=2 hlim=64 time=4.000 ms
56 bytes from 3003::1, icmp_seq=3 hlim=64 time=10.000 ms
56 bytes from 3003::1, icmp_seq=4 hlim=64 time=11.000 ms

--- Ping6 statistics for 3003::1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 4.000/16.000/45.000/14.711 ms
```

Configuring an automatic IPv4-compatible IPv6 tunnel

Restrictions and guidelines

Follow these guidelines when you configure an automatic IPv4-compatible IPv6 tunnel:

- You do not need to configure a destination address for an automatic IPv4-compatible IPv6 tunnel. The destination address of the tunnel is embedded in the destination IPv4-compatible IPv6 address.
- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.

Procedure

1. Enter system view.

   `system-view`

2. Enter automatic IPv4-compatible IPv6 tunnel interface view.

   `interface tunnel` *`number`* [**`mode ipv6-ipv4 auto-tunnel`**]

3. Specify an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."

4. Configure a source address or source interface for the tunnel.

   `source` { *`ipv4-address`* | *`interface-type interface-number`* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

5. (Optional.) Set the DF bit for tunneled packets.

   `tunnel dfbit enable`

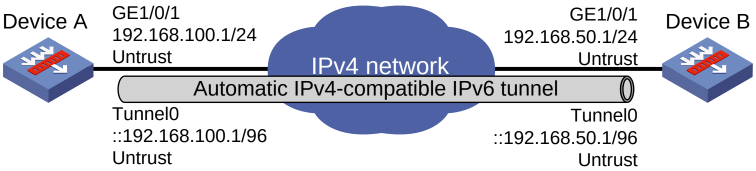
   By default, the DF bit is not set for tunneled packets.

Example: Configuring an automatic IPv4-compatible IPv6 tunnel

Network configuration

As shown in Figure 7, dual-stack devices Device A and Device B communicate over an IPv4 network. Configure an automatic IPv4-compatible IPv6 tunnel between the two devices to enable IPv6 communications over the IPv4 network.

Figure 7 Network diagram



Procedure

1. Configure Device A:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 192.168.100.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Create an automatic IPv4-compatible IPv6 tunnel interface.

```
[DeviceA] interface tunnel 0 mode ipv6-ipv4 auto-tunnel
[DeviceA-Tunnel0] ipv6 address ::192.168.100.1/96
[DeviceA-Tunnel0] source gigabitethernet 1/0/1
```

   # Configure settings for routing. This example configures a static route, and the next hop in the route is 192.168.100.2.

```
[DeviceA] ip route-static 192.168.50.1 24 192.168.100.2
```

   # Add interfaces to security zone **Untrust**.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
```

   # Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 192.168.100.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 192.168.50.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
```

```
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalout
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-ip-host ::192.168.100.1
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-ip-host ::192.168.50.1
[DeviceA-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceA-security-policy-ipv6-1-tunnellocalout] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 192.168.50.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 192.168.100.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalin
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-ip-host ::192.168.50.1
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-ip-host ::192.168.100.1
[DeviceA-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceA-security-policy-ipv6-2-tunnellocalint] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.50.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Create an automatic IPv4-compatible IPv6 tunnel interface.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4 auto-tunnel
[DeviceB-Tunnel0] ipv6 address ::192.168.50.1/96
[DeviceB-Tunnel0] source gigabitethernet 1/0/1
```

# Configure settings for routing. This example configures a static route, and the next hop in the route is 192.168.50.2.

```
[DeviceB] ip route-static 192.168.100.1 24 192.168.50.2
```

# Add interfaces to security zone **Untrust**.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 192.168.50.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 192.168.100.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalout
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-ip-host ::192.168.50.1
[DeviceB-security-policy-ipv6-1-tunnellocalout]
destination-ip-host ::192.168.100.1
[DeviceB-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceB-security-policy-ipv6-1-tunnellocalout] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 192.168.100.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 192.168.50.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-ip-host ::192.168.100.1
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-ip-host ::192.168.50.1
[DeviceB-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-2-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Use the **display ipv6 interface** command to display tunnel interface status on Device A and Device B. Verify that interface Tunnel 0 is up. (Details not shown.)

# Verify that Device B and Device A can ping the IPv4-compatible IPv6 address of each other. This example uses Device A.

```
[DeviceA-Tunnel0] ping ipv6 ::192.168.50.1
Ping6(56 data bytes) ::192.168.100.1 --> ::192.168.50.1, press CTRL_C to break
56 bytes from ::192.168.50.1, icmp_seq=0 hlim=64 time=17.000 ms
56 bytes from ::192.168.50.1, icmp_seq=1 hlim=64 time=9.000 ms
56 bytes from ::192.168.50.1, icmp_seq=2 hlim=64 time=11.000 ms
56 bytes from ::192.168.50.1, icmp_seq=3 hlim=64 time=9.000 ms
56 bytes from ::192.168.50.1, icmp_seq=4 hlim=64 time=11.000 ms

--- Ping6 statistics for ::192.168.50.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 9.000/11.400/17.000/2.939 ms
```

Configuring a 6to4 tunnel

Restrictions and guidelines

Follow these guidelines when you configure a 6to4 tunnel:

- You do not need to configure a destination address for a 6to4 tunnel, because the destination IPv4 address is embedded in the 6to4 IPv6 address.
- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.
- Automatic tunnels do not support dynamic routing. You must configure a static route destined for the destination IPv6 network if the destination IPv6 network is not in the same subnet as the IPv6 address of the tunnel interface. You can specify the local tunnel interface as the egress interface of the route or specify the IPv6 address of the peer tunnel interface as the next hop of the route. For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter 6to4 tunnel interface view.

   interface tunnel *number* [**mode ipv6-ipv4 6to4**]

3. Specify an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."

4. Configure a source address or source interface for the tunnel.

   source { *ipv4-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

5. (Optional.) Set the DF bit for tunneled packets.

```
tunnel dfbit enable
```

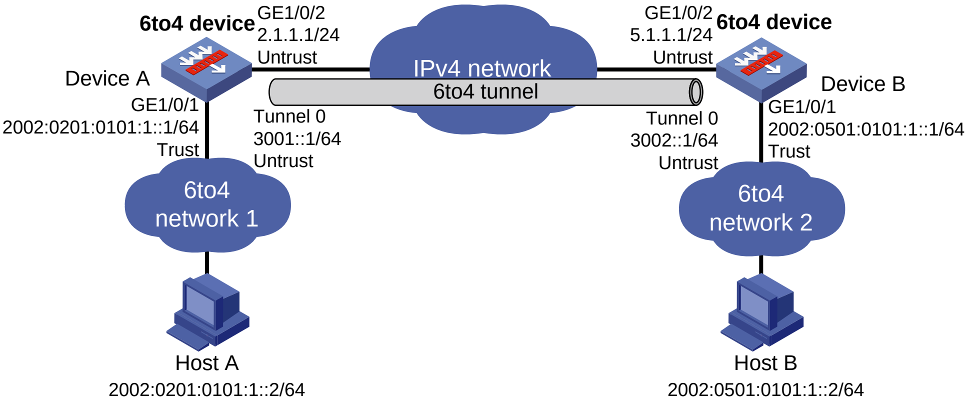
By default, the DF bit is not set for tunneled packets.

Example: Configuring a 6to4 tunnel

Network configuration

As shown in Figure 8, configure a 6to4 tunnel between 6to4 devices Device A and Device B so the two hosts can reach each other over the IPv4 network.

Figure 8 Network diagram



Analysis

To enable communication between 6to4 networks, configure 6to4 addresses for 6to4 devices and hosts in the 6to4 networks.

- The IPv4 address of GigabitEthernet 1/0/2 on Device A is 2.1.1.1/24, and the corresponding 6to4 prefix is 2002:0201:0101::/48. Host A must use this prefix.
- The IPv4 address of GigabitEthernet 1/0/2 on Device B is 5.1.1.1/24, and the corresponding 6to4 prefix is 2002:0501:0101::/48. Host B must use this prefix.

Procedure

1. Configure Device A:

   # Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 2002:0201:0101:1::1/64
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 2.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/2] quit
```

   # Create 6to4 tunnel interface **Tunnel 0**.

```
[DeviceA] interface tunnel 0 mode ipv6-ipv4 6to4
[DeviceA-Tunnel0] ipv6 address 3001::1/64
[DeviceA-Tunnel0] source gigabitethernet 1/0/2
[DeviceA-Tunnel0] quit
```

   # Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 2.1.1.2.

```
[DeviceA] ip route-static 5.1.1.1 24 2.1.1.2
[DeviceA] ipv6 route-static 2002:0501:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 2.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 5.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2002:0201:: 32
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-ip-subnet 2002:0501::
32
[DeviceA-security-policy-ipv6-1-trust-untrust] action pass
[DeviceA-security-policy-ipv6-1-trust-untrust] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 5.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 2.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2002:0501:: 32
```

```
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-ip-subnet 2002:0201::
32
[DeviceA-security-policy-ipv-2-untrust-trust] action pass
[DeviceA-security-policy-ipv6-2-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2002:0501:0101:1::1/64
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 5.5.5.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create 6to4 tunnel interface **Tunnel 0**.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4 6to4
[DeviceB-Tunnel0] ipv6 address 3002::1/64
[DeviceB-Tunnel0] source gigabitethernet 1/0/2
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 5.1.1.2.

```
[DeviceB] ip route-static 2.1.1.1 24 5.1.1.2
[DeviceB] ipv6 route-static 2002:0201:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 5.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 2.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-zone untrust
```

```
[DeviceB-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2002:0501:: 32
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-ip-subnet 2002:0201::
32
[DeviceB-security-policy-ipv6-1-trust-untrust] action pass
[DeviceB-security-policy-ipv6-1-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 2.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 5.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-ip-subnet 2002:0201:: 32
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-ip-subnet 2005:0201::
32
[DeviceB-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-2-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Verify that Linux-running hosts Host A and Host B can ping each other.

```
D:\>ping6 -s 2002:201:101:1::2 2002:501:101:1::2

Pinging 2002:501:101:1::2
from 2002:201:101:1::2 with 32 bytes of data:

Reply from 2002:501:101:1::2: bytes=32 time=13ms
Reply from 2002:501:101:1::2: bytes=32 time=1ms
Reply from 2002:501:101:1::2: bytes=32 time=1ms
Reply from 2002:501:101:1::2: bytes=32 time<1ms

Ping statistics for 2002:501:101:1::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 3ms
```

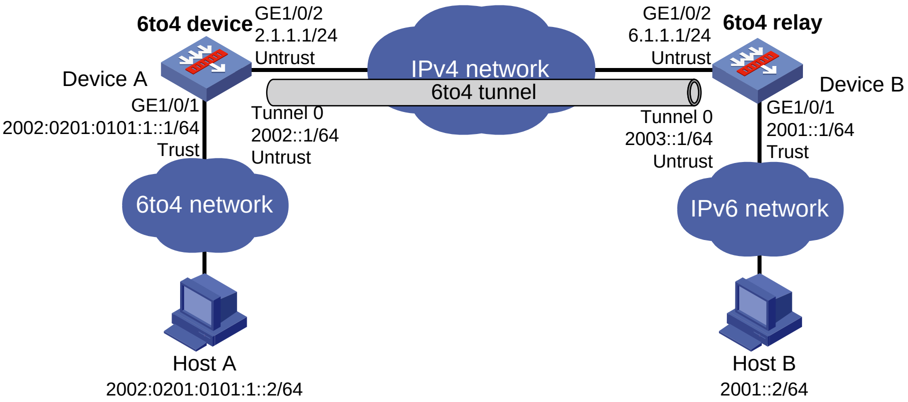
Example: Configuring 6to4 relay

Network configuration

As shown in Figure 9, Device A is a 6to4 device, and 6to4 addresses are used on the connected IPv6 network. Device B acts as a 6to4 relay device and is connected to an IPv6 network (2001::/16). Configure a 6to4 tunnel between Device A and Device B to make Host A and Host B reachable to each other.

The configuration on a 6to4 relay device is similar to that on a 6to4 device. However, to enable communication between the 6to4 network and the IPv6 network, you must configure a route to the IPv6 network on the 6to4 device. The IPv4 address of GigabitEthernet 1/0/2 on the relay device is 6.1.1.1/24 and its corresponding 6to4 prefix is 2002:0601:0101::/48. The next hop of the static route must be an address using this prefix.

Figure 9 Network diagram



Procedure

1. Configure Device A:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 2002:0201:0101:1::1/64
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 2.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/2] quit
```

# Create 6to4 tunnel interface **Tunnel 0**.

```
[DeviceA] interface tunnel 0 mode ipv6-ipv4 6to4
[DeviceA-Tunnel0] ipv6 address 2002::1/64
[DeviceA-Tunnel0] source gigabitethernet 1/0/2
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. The route with a destination prefix of **::/0**. is a default route, and the next hop is the 6to4 address of the relay device. In the other routes, the output interface is Tunnel 0 and the next hop is 2.1.1.2.

```
[DeviceA] ip route-static 6.1.1.1 24 2.1.1.2
[DeviceA] ipv6 route-static 2002:0601:0101:: 64 tunnel 0
[DeviceA] ipv6 route-static :: 0 2002:0601:0101::1
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
```

```
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 2.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 6.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2002:0201:0101:
1:: 64
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-ip-subnet 2001:: 64
[DeviceA-security-policy-ipv6-1-trust-untrust] action pass
[DeviceA-security-policy-ipv6-1-trust-untrust] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 6.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 2.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2001:: 64
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-ip-subnet
2002:0201:0101:
```

```
1:: 64
[DeviceA-security-policy-ipv-2-untrust-trust] action pass
[DeviceA-security-policy-ipv6-2-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2001::1/64
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 6.1.1.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create 6to4 tunnel interface **Tunnel 0**.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4 6to4
[DeviceB-Tunnel0] ipv6 address 2003::1/64
[DeviceB-Tunnel0] source gigabitethernet 1/0/2
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 6.1.1.2.

```
[DeviceB] ip route-static 2.1.1.1 24 6.1.1.2
[DeviceB] ipv6 route-static 2002:: 16 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 6.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 2.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2001:: 64
```

```
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-ip-subnet
2002:0201:0101:
1:: 64
[DeviceB-security-policy-ipv6-1-trust-untrust] action pass
[DeviceB-security-policy-ipv6-1-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 2.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 6.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name untrust-trust
[DeviceB-security-policy-ipv6-2-untrust-trust] source-zone untrust
[DeviceB-security-policy-ipv6-2-untrust-trust] destination-zone trust
[DeviceB-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2002:0201:0101:
1:: 64
[DeviceB-security-policy-ipv6-2-untrust-trust] destination-ip-subnet 2001:: 64
[DeviceB-security-policy-ipv-2-untrust-trust] action pass
[DeviceB-security-policy-ipv6-2-untrust-trust] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Verify that Linux-running hosts Host A and Host B can ping each other.

```
D:\>ping6 -s 2002:201:101:1::2 2001::2

Pinging 2001::2
from 2002:201:101:1::2 with 32 bytes of data:

Reply from 2001::2: bytes=32 time=13ms
Reply from 2001::2: bytes=32 time=1ms
Reply from 2001::2: bytes=32 time=1ms
Reply from 2001::2: bytes=32 time<1ms

Ping statistics for 2001::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 3ms
```

Configuring an ISATAP tunnel

Restrictions and guidelines

Follow these guidelines when you configure an ISATAP tunnel:

- You do not need to configure a destination address for an ISATAP tunnel, because the destination IPv4 address is embedded in the ISATAP address.
- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.
- Because automatic tunnels do not support dynamic routing, configure a static route destined for the destination IPv6 network at each tunnel end. You can specify the local tunnel interface as the egress interface of the route or specify the IPv6 address of the peer tunnel interface as the next hop of the route. For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter ISATAP tunnel interface view.

   interface tunnel *number* [**mode ipv6-ipv4 isatap**]

3. Specify an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."

4. Configure a source address or source interface for the tunnel.

   source { *ipv4-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

5. (Optional.) Set the DF bit for tunneled packets.

   tunnel dfbit enable

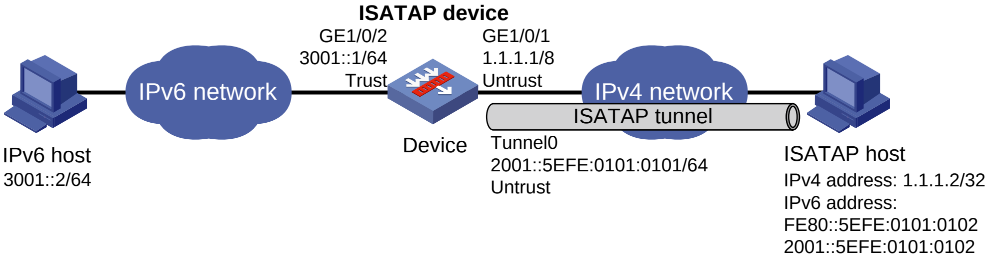
   By default, the DF bit is not set for tunneled packets.

Example: Configuring an ISATAP tunnel

Network configuration

As shown in Figure 10, configure an ISATAP tunnel between the device and the ISATAP host so the ISATAP host in the IPv4 network can access the IPv6 network.

Figure 10 Network diagram



Procedure

1. Configure the device:

   # Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 1.1.1.1 8
   [Device-GigabitEthernet1/0/1] quit
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] ipv6 address 3001::1/64
   [Device-GigabitEthernet1/0/2] quit
   ```

   # Create ISATAP tunnel interface **Tunnel 0**.

   ```
   [Device] interface tunnel 0 mode ipv6-ipv4 isatap
   [Device-Tunnel0] ipv6 address 2001:: 64 eui-64
   [Device-Tunnel0] source gigabitethernet 1/0/1
   [Device-Tunnel0] undo ipv6 nd ra halt
   [Device-Tunnel0] quit
   ```

   # Add interfaces to security zones.

   ```
   [Device] security-zone name Untrust
   [Device-security-zone-Untrust] import interface Tunnel 0
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] quit
   [Device] security-zone name Trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Trust] quit
   ```

   # Configure a rule named **tunnellocalout** in the IPv4 security policy to allow the device to send packets to the ISATAP host.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name tunnellocalout
   [Device-security-policy-ip-1-tunnellocalout] source-zone local
   [Device-security-policy-ip-1-tunnellocalout] destination-zone untrust
   [Device-security-policy-ip-1-tunnellocalout] source-ip-host 1.1.1.1
   [Device-security-policy-ip-1-tunnellocalout] destination-ip-host 1.1.1.2
   [Device-security-policy-ip-1-tunnellocalout] action pass
   [Device-security-policy-ip-1-tunnellocalout] quit
   [Device-security-policy-ip] quit
   ```

   # Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from the IPv6 host to the ISATAP host.

   ```
   [Device] security-policy ipv6
   [Device-security-policy-ipv6] rule name trust-untrust
   [Device-security-policy-ipv6-1-trust-untrust] source-zone trust
   [Device-security-policy-ipv6-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ipv6-1-trust-untrust] source-ip-subnet 3001:: 64
   [Device-security-policy-ipv6-1-trust-untrust] destination-ip-subnet 2001:: 64
   [Device-security-policy-ipv6-1-trust-untrust] action pass
   [Device-security-policy-ipv6-1-trust-untrust] quit
   [Device-security-policy-ipv6] quit
   ```

   # Configure a rule named **tunnellocalin** in the IPv4 security policy to allow the device to receive the packets sent from the ISATAP host.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name tunnellocalin
[Device-security-policy-ip-2-tunnellocalin] source-zone untrust
[Device-security-policy-ip-2-tunnellocalin] destination-zone local
[Device-security-policy-ip-2-tunnellocalin] source-ip-host 1.1.1.2
[Device-security-policy-ip-2-tunnellocalin] destination-ip-host 1.1.1.1
[Device-security-policy-ip-2-tunnellocalin] action pass
[Device-security-policy-ip-2-tunnellocalin] quit
[Device-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from the ISATAP host to the IPv6 host.

```
[Device] security-policy ipv6
[Device-security-policy-ipv6] rule name untrust-trust
[Device-security-policy-ipv6-2-untrust-trust] source-zone untrust
[Device-security-policy-ipv6-2-untrust-trust] destination-zone trust
[Device-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2001:: 64
[Device-security-policy-ipv6-2-untrust-trust] destination-ip-subnet 3001:: 64
[Device-security-policy-ipv6-2-untrust-trust] action pass
[Device-security-policy-ipv6-2-untrust-trust] quit
[Device-security-policy-ipv6] quit
```

2. Configure the ISATAP host:

   Settings on the ISATAP host vary by operating system. The following configuration is performed on Windows XP.

   # Install IPv6.

```
C:\>ipv6 install
```

   # On a host running Windows XP, the ISATAP interface is typically interface 2. Display information about the ISATAP interface.

```
C:\>ipv6 if 2
Interface 2: Automatic Tunneling Pseudo-Interface
  Guid {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
  does not use Neighbor Discovery
  does not use Router Discovery
  routing preference 1
  EUI-64 embedded IPv4 address: 0.0.0.0
  router link-layer address: 0.0.0.0
  preferred link-local fe80::5efe:1.1.1.2, life infinite
  link MTU 1280 (true link MTU 65515)
  current hop limit 128
  reachable time 42500ms (base 30000ms)
  retransmission interval 1000ms
  DAD transmits 0
  default site prefix length 48
```

   # Specify an IPv4 address for the ISATAP device.

```
C:\>netsh interface ipv6 isatap set router 1.1.1.1
```

   # Display information about the ISATAP interface.

```
C:\>ipv6 if 2
Interface 2: Automatic Tunneling Pseudo-Interface
  Guid {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
```

```
    does not use Neighbor Discovery
    uses Router Discovery
    routing preference 1
    EUI-64 embedded IPv4 address: 1.1.1.2
    router link-layer address: 1.1.1.1
    preferred global 2001::5efe:1.1.1.2, life 29d23h59m46s/6d23h59m46s (public)
    preferred link-local fe80::5efe:1.1.1.2, life infinite
    link MTU 1500 (true link MTU 65515)
    current hop limit 255
    reachable time 42500ms (base 30000ms)
    retransmission interval 1000ms
    DAD transmits 0
    default site prefix length 48
```

The host has obtained the prefix 2001::/64 and has automatically generated the global unicast address 2001::5efe:1.1.1.2. The message "uses Router Discovery" indicates that the router discovery feature is enabled on the host.

# Display information about IPv6 routes on the host.

```
C:\>ipv6 rt
2001::/64 -> 2 pref 1if+8=9 life 29d23h59m43s (autoconf)
::/0 -> 2/fe80::5efe:1.1.1.1 pref 1if+256=257 life 29m43s (autoconf)
```

3. On the IPv6 host, configure a route to the ISATAP device.

```
C:\>netsh interface ipv6 set route 2001::/64 5 3001::1
```

Verifying the configuration

# Verify that the ISATAP host can ping the IPv6 host.

```
C:\>ping 3001::2

Pinging 3001::2 with 32 bytes of data:

Reply from 3001::2: time=1ms
Reply from 3001::2: time=1ms
Reply from 3001::2: time=1ms
Reply from 3001::2: time=1ms

Ping statistics for 3001::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Configuring a 6RD tunnel

Hardware compatibility with 6RD tunneling

Models	6RD tunneling compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

NFNX3-HDB680, NFNX3-HDB1080	Yes

Restrictions and guidelines

Follow these guidelines when you configure a 6RD tunnel:

- You do not need to configure a destination address for a 6RD tunnel. The device automatically identifies the destination IPv4 address according to the IPv4 address embedded in the 6RD address, the 6RD prefix, and the IPv4 prefix and suffix.
- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.
- Automatic tunnels do not support dynamic routing. You must configure a static route destined for the destination IPv6 network if the destination IPv6 network is not in the same subnet as the IPv6 address of the tunnel interface. You can specify the local tunnel interface as the egress interface of the route or specify the IPv6 address of the peer tunnel interface as the next hop of the route. For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter 6RD tunnel interface view.

   ```
   interface tunnel number mode ipv6-ipv4 6rd
   ```

3. Specify an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."

4. Configure a source address or source interface for the tunnel.

   ```
   source { ipv4-address | interface-type interface-number }
   ```

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.

5. Configure the 6RD prefix for the tunnel.

   ```
   tunnel 6rd prefix ipv6-prefix/prefix-length
   ```

   By default, no 6RD prefix is configured for the tunnel.

6. (Optional.) Specify a prefix length and suffix length for the tunnel source address.

   ```
   tunnel 6rd ipv4 { prefix-length length | suffix-length length } *
   ```

   By default, all 32 bits of the IPv4 tunnel source address are used to create the 6RD delegated prefix.

7. (Optional.) Specify a BR address for the tunnel.

   ```
   tunnel 6rd br ipv4-address
   ```

   By default, no BR address is specified for a 6RD tunnel.

8. (Optional.) Set the DF bit for tunneled packets.

   ```
   tunnel dfbit enable
   ```

   By default, the DF bit is not set for tunneled packets.

Display and maintenance commands for 6RD tunneling

Execute **display** commands in any view.

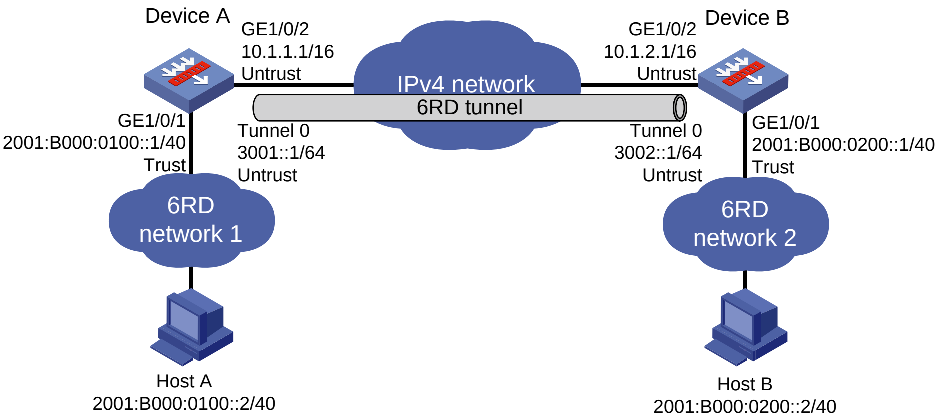
Task	Command
Display information about 6RD tunnel interfaces.	**display 6rd** [**interface tunnel** *number*]
Display a 6RD tunnel destination address.	**display 6rd destination prefix** *ipv6-prefix* **interface tunnel** *number*
Display a 6RD delegated prefix.	**display 6rd prefix destination** *ipv4-address* **interface tunnel** *number*

Example: Configuring a 6RD tunnel

Network configuration

As shown in Figure 11, Host A and Host B are in different 6RD networks. Configure a 6RD tunnel between Device A and Device B so Host A and Host B can reach each other over the IPv4 network.

Figure 11 Network diagram



Requirements analysis

To enable communication between 6RD networks, you must perform the following tasks in addition to the 6RD tunnel configuration:

- On Device A and Device B, configure the 6RD prefix as 2001:B000::/32, the IPv4 prefix length as 16, and the IPv4 suffix length as 8.
- Configure 6RD addresses for Device A, Device B, and the hosts.
    - On Device A, the tunnel source IPv4 address is 10.1.1.1/16, the IPv4 address of GigabitEthernet 1/0/2. Calculated based on this IPv4 address, the 6RD prefix, and the IPv4 prefix and suffix lengths, the 6RD delegated prefix for 6RD network 1 is 2001:B000:100::/40. The 6RD addresses of Device A and Host A must use this prefix.
    - On Device B, the tunnel source IPv4 address is 10.1.2.1/16, the IPv4 address of GigabitEthernet 1/0/2. Calculated based on this IPv4 address, the 6RD prefix, and the IPv4 prefix and suffix lengths, the 6RD delegated prefix for 6RD network 2 is 2001:B000:200::/40. The 6RD addresses of Device B and Host B must use this prefix.

Procedure

Make sure Device A and Device B can reach each other through IPv4.

1. Configure Device A:

# Assign an IPv4 address to interface GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 10.1.1.1 16
[DeviceA-GigabitEthernet1/0/2] quit
```

# Assign a 6RD address to interface GigabitEthernet 1/0/1.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 2001:b000:0100::1/40
[DeviceA-GigabitEthernet1/0/1] quit
```

# Create 6RD tunnel interface **Tunnel 0**.

```
[DeviceA] interface tunnel 0 mode ipv6-ipv4 6rd
```

# Assign an IPv6 address to the tunnel interface.

```
[DeviceA-Tunnel0] ipv6 address 3001::1/64
```

# Specify GigabitEthernet 1/0/2 as the source interface of the tunnel.

```
[DeviceA-Tunnel0] source gigabitethernet 1/0/2
```

# Configure the 6RD prefix of the tunnel.

```
[DeviceA-Tunnel0] tunnel 6rd prefix 2001:b000::/32
```

# Specify a prefix length and a suffix length for the tunnel.

```
[DeviceA-Tunnel0] tunnel 6rd ipv4 prefix-len 16 suffix-len 8
[DeviceA-Tunnel0] quit
```

# Configure a static route destined for 2001:B000::/32 through the tunnel.

```
[DeviceA] ipv6 route-static 2001:b000:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 10.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 10.1.2.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-zone untrust
```

```
[DeviceA-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2001:B000::0100::
40
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-ip-subnet
2001:B000::0200:: 40
[DeviceA-security-policy-ipv6-1-trust-untrust] action pass
[DeviceA-security-policy-ipv6-1-trust-untrust] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 10.1.2.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 10.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2001:B000::0200::
40
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-ip-subnet
2001:B000::0100:: 40
[DeviceA-security-policy-ipv6-2-untrust-trust] action pass
[DeviceA-security-policy-ipv6-2-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign an IPv4 address to interface GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 10.1.2.1 16
[DeviceB-GigabitEthernet1/0/2] quit
```

# Assign a 6RD address to interface GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2001:b000:0200::1/40
[DeviceB-GigabitEthernet1/0/1] quit
```

# Create 6RD tunnel interface **Tunnel 0**.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4 6rd
```

# Assign an IPv6 address to the tunnel interface.

```
[DeviceB-Tunnel0] ipv6 address 3002::1/64
```

# Specify GigabitEthernet 1/0/2 as the source interface of the tunnel.

```
[DeviceB-Tunnel0] source gigabitethernet 1/0/2
```

# Configure the 6RD prefix of the tunnel.

```
[DeviceB-Tunnel0] tunnel 6rd prefix 2001:b000::/32
```

# Specify a prefix length and a suffix length for the tunnel.

```
[DeviceB-Tunnel0] tunnel 6rd ipv4 prefix-len 16 suffix-len 8
[DeviceB-Tunnel0] quit
```

# Configure a static route destined for 2001:B000::/32 through the tunnel.

```
[DeviceB] ipv6 route-static 2001:b000:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 10.1.2.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2001:B000::0200::
40
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-ip-subnet
2001:B000::0100:: 40
[DeviceB-security-policy-ipv6-1-trust-untrust] action pass
[DeviceB-security-policy-ipv6-1-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 10.1.2.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-ip-subnet 2001:B000::0100::
40
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-ip-subnet
2001:B000::0200:: 40
[DeviceB-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-2-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Verify that Host A and Host B can ping each other.

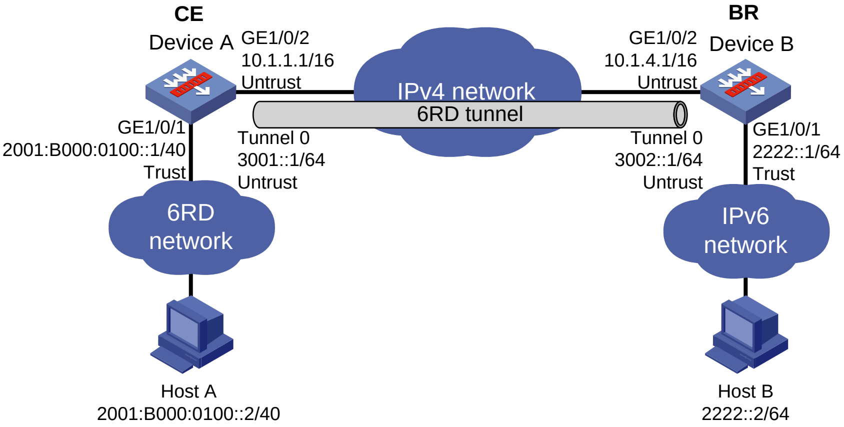
```
D:\>ping6 -s 2001:b000:0100::2 2001:b000:0200::2

Pinging 2001:B000:0200::2
from 2001:B000:0100::2 with 32 bytes of data:

Reply from 2001:B000:0200::2: bytes=32 time=13ms
Reply from 2001:B000:0200::2: bytes=32 time=1ms
Reply from 2001:B000:0200::2: bytes=32 time=1ms
Reply from 2001:B000:0200::2: bytes=32 time<1ms

Ping statistics for 2001:B000:0200::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 3ms
```

Example: Configuring 6RD relay

Network configuration

As shown in Figure 12, Host A is in a 6RD network connected to Device A (a 6RD CE). Host B is in a common IPv6 network connected to Device B (a 6RD BR router). Configure a 6RD tunnel between Device A and Device B to make Host A and Host B reachable to each other.

Figure 12 Network diagram



Requirements analysis

To enable communication between the 6RD network and the IPv6 network, you must perform the following tasks in addition to the 6RD tunnel configuration:

- On Device A and Device B, configure the 6RD prefix as 2001:B000::/32, the IPv4 prefix length as 16, and the IPv4 suffix length as 8.
- Configure 6RD addresses for Device A and Host A in the 6RD network.

  On Device A, the tunnel source IPv4 address is 10.1.1.1/16, the IPv4 address of GigabitEthernet 1/0/2. Calculated based on this IPv4 address, the 6RD prefix, and the IPv4 prefix and suffix lengths, the 6RD delegated prefix for the 6RD network is 2001:B000:100::/40. The 6RD addresses of Device A and Host A must use this prefix.
- On Device A, you must configure a static route to the IPv6 network. The next hop address of the route is a 6RD address of Device B.

  On Device B, the tunnel source IPv4 address is 10.1.4.1/16, the IPv4 address of GigabitEthernet 1/0/2. Calculated based on this IPv4 address, the 6RD prefix, and the IPv4 prefix and suffix lengths, the 6RD delegated prefix is 2001:B000:0400::/40. The next hop address of the static route can be any address using this prefix.

Procedure

Make sure Device A and Device B can reach each other through IPv4.

1. Configure Device A:

   # Assign an IPv4 address to interface GigabitEthernet 1/0/2.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/2
   [DeviceA-GigabitEthernet1/0/2] ip address 10.1.1.1 16
   [DeviceA-GigabitEthernet1/0/2] quit
   ```

   # Assign a 6RD address to interface GigabitEthernet 1/0/1.

   ```
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ipv6 address 2001:b000:0100::1/40
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Create 6RD tunnel interface **Tunnel 0**.

   ```
   [DeviceA] interface tunnel 0 mode ipv6-ipv4 6rd
   ```

   # Assign an IPv6 address to the tunnel interface.

   ```
   [DeviceA-Tunnel0] ipv6 address 3001::1/64
   ```

   # Specify GigabitEthernet 1/0/2 as the source interface of the tunnel.

   ```
   [DeviceA-Tunnel0] source gigabitethernet 1/0/2
   ```

# Configure the 6RD prefix of the tunnel.

```
[DeviceA-Tunnel0] tunnel 6rd prefix 2001:b000::/32
```

# Specify a prefix length and a suffix length for the tunnel.

```
[DeviceA-Tunnel0] tunnel 6rd ipv4 prefix-len 16 suffix-len 8
```

# Specify the BR address on the tunnel interface.

```
[DeviceA-Tunnel0] tunnel 6rd br 10.1.4.1
[DeviceA-Tunnel0] quit
```

# Configure a default route to reach the IPv6 network, which specifies the next hop as the 6RD address of Device B.

```
[DeviceA] ipv6 route-static :: 0 2001:b000:0400::1
```

# Configure a static route destined for 2001:B000::/32 through the tunnel.

```
[DeviceA] ipv6 route-static 2001:b000:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 10.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 10.1.4.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2001:B000::0100::
40
[DeviceA-security-policy-ipv6-1-trust-untrust] destination-ip-subnet 2222:: 64
[DeviceA-security-policy-ipv6-1-trust-untrust] action pass
[DeviceA-security-policy-ipv6-1-trust-untrust] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-2-tunnellocalin] source-zone untrust
```

```
[DeviceA-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-2-tunnellocalin] source-ip-host 10.1.4.1
[DeviceA-security-policy-ip-2-tunnellocalin] destination-ip-host 10.1.1.1
[DeviceA-security-policy-ip-2-tunnellocalin] action pass
[DeviceA-security-policy-ip-2-tunnellocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-2-untrust-trust] source-ip-subnet 2222:: 64
[DeviceA-security-policy-ipv6-2-untrust-trust] destination-ip-subnet
2001:B000::0100:: 40
[DeviceA-security-policy-ipv6-2-untrust-trust] action pass
[DeviceA-security-policy-ipv6-2-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign an IPv4 address to interface GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 10.1.4.1 16
[DeviceB-GigabitEthernet1/0/2] quit
```

# Assign an IPv6 address to interface GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2222::1/64
[DeviceB-GigabitEthernet1/0/1] quit
```

# Create 6RD tunnel interface **Tunnel 0**.

```
[DeviceB] interface tunnel 0 mode ipv6-ipv4 6rd
```

# Assign an IPv6 address to the tunnel interface.

```
[DeviceB-Tunnel0] ipv6 address 3002::1/64
```

# Specify GigabitEthernet 1/0/2 as the source interface of the tunnel.

```
[DeviceB-Tunnel0] source gigabitethernet 1/0/2
```

# Configure the 6RD prefix of the tunnel.

```
[DeviceB-Tunnel0] tunnel 6rd prefix 2001:b000::/32
```

# Specify a prefix length and a suffix length for the tunnel.

```
[DeviceB-Tunnel0] tunnel 6rd ipv4 prefix-len 16 suffix-len 8
[DeviceB-Tunnel0] quit
```

# Configure a static route destined for 2001:B000::/32 through the tunnel.

```
[DeviceB] ipv6 route-static 2001:b000:: 32 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
```

```
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 10.1.4.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from Host B to Host A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-1-trust-untrust] source-ip-subnet 2222:: 64
[DeviceB-security-policy-ipv6-1-trust-untrust] destination-ip-subnet
2001:B000::0100:: 40
[DeviceB-security-policy-ipv6-1-trust-untrust] action pass
[DeviceB-security-policy-ipv6-1-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-2-tunnellocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-2-tunnellocalin] destination-ip-host 10.1.4.1
[DeviceB-security-policy-ip-2-tunnellocalin] action pass
[DeviceB-security-policy-ip-2-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to permit the packets from Host A to Host B.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-2-tunnellocalin] source-ip-subnet 2001:B000::0100::
40
[DeviceB-security-policy-ipv6-2-tunnellocalin] destination-ip-subnet 2222:: 64
[DeviceB-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-2-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Verify that Host A and Host B can ping each other.

```
D:\>ping6 -s 2001:b000:0100::2 2222::2

Pinging 2222::2
from 2001:B000:0100::2 with 32 bytes of data:

Reply from 2222::2: bytes=32 time=13ms
Reply from 2222::2: bytes=32 time=1ms
Reply from 2222::2: bytes=32 time=1ms
Reply from 2222::2: bytes=32 time<1ms

Ping statistics for 2222::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 3ms
```

Enabling dropping IPv6 packets that use IPv4-compatible IPv6 addresses

Restrictions and guidelines

Automatic IPv4-compatible IPv6 tunnels do not support this feature.

Procedure

1. Enter system view.
   `system-view`
2. Enable dropping IPv6 packets that use IPv4-compatible IPv6 addresses.
   `tunnel discard ipv4-compatible-packet`
   By default, IPv6 packets that use IPv4-compatible IPv6 addresses are not dropped.

IPv4 over IPv4 tunneling

About IPv4 over IPv4 tunneling

IPv4 over IPv4 tunneling (RFC 1853) enables isolated IPv4 networks to communicate. For example, an IPv4 over IPv4 tunnel can connect isolated private IPv4 networks over a public IPv4 network.

Figure 13 IPv4 over IPv4 tunnel

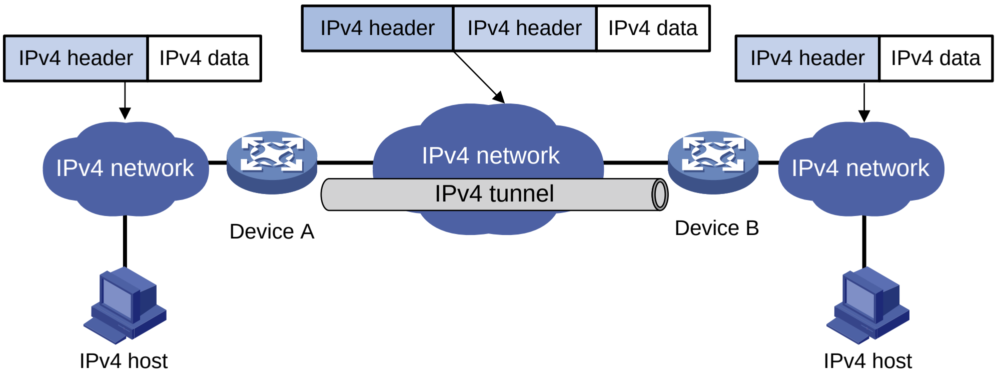


Figure 13 shows the encapsulation and de-encapsulation processes.

- Encapsulation:
  - **a.** Device A receives an IP packet from an IPv4 host and submits it to the IP protocol stack.
  - **b.** The IPv4 protocol stack determines how to forward the packet according to the destination address in the IP header. If the packet is destined for the IPv4 host connected to Device B, Device A delivers the packet to the tunnel interface.
  - **c.** The tunnel interface adds a new IPv4 header to the IPv4 packet and submits it to the IP protocol stack.

    In the new header, the source IP address specifies the tunnel source, and the destination IP address specifies the tunnel destination.
  - **d.** The IP protocol stack uses the destination IP address of the new IP header to look up the routing table, and then sends the packet out.
- De-encapsulation:
  - **a.** After receiving the packet, Device B delivers it to the IP protocol stack.
  - **b.** If the protocol number is 4 (indicating an IPv4 packet is encapsulated within the packet), the IP protocol stack delivers the packet to the tunnel module for de-encapsulation.
  - **c.** The tunnel module de-encapsulates the IP packet and sends it back to the IP protocol stack.
  - **d.** The protocol stack forwards the de-encapsulated packet.

Restrictions and guidelines: IPv4 over IPv4 tunnel configuration

Follow these guidelines when you configure an IPv4 over IPv4 tunnel:

- The tunnel destination address specified on the local device must be identical with the tunnel source address specified on the tunnel peer device.

- Do not specify the same tunnel source and destination addresses for tunnels of the same tunnel mode.
- The IPv4 address of the local tunnel interface cannot be on the same subnet as the destination address configured on the tunnel interface.
- To ensure correct packet forwarding, identify whether the destination IPv4 network and the IPv4 address of the local tunnel interface are on the same subnet. If they are not, configure a route reaching the destination IPv4 network through the tunnel interface. You can configure the route by using one of the following methods:
  - Configure a static route, and specify the local tunnel interface as the egress interface or specify the IPv4 address of the peer tunnel interface as the next hop.
  - Enable a dynamic routing protocol on both the local and remote tunnel interfaces.

  For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.
- The destination address of the route passing the tunnel interface cannot be on the same subnet as the destination address configured on the tunnel interface.

Configuring an IPv4 over IPv4 tunnel

1. Enter system view.

   system-view
2. Enter IPv4 over IPv4 tunnel interface view.

   interface tunnel *number* [**mode ipv4-ipv4**]
3. Configure an IPv4 address for the tunnel interface.

   ip address *ip-address* { *mask* | *mask-length* } [**sub**]
4. Configure a source address or source interface for the tunnel.

   source { *ipv4-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel.

   If you specify a source address, it is used as the source IP address of tunneled packets.

   If you specify a source interface, the primary IP address of this interface is used as the source IP address of tunneled packets.
5. Configure a destination address for the tunnel.

   destination *ipv4-address*

   By default, no destination address is configured for a tunnel.

   The tunnel destination address must be the IP address of the receiving interface on the tunnel peer. It is used as the destination IP address of tunneled packets.
6. (Optional.) Set the DF bit for tunneled packets.

   tunnel dfbit enable

   By default, the DF bit is not set for tunneled packets.

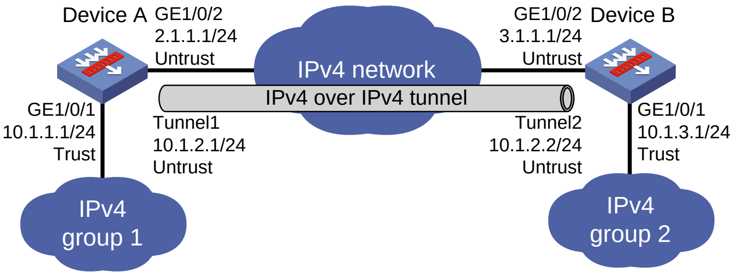
IPv4 over IPv4 tunnel configuration examples

Example: Configuring an IPv4 over IPv4 tunnel

Network configuration

As shown in Figure 14, the two subnets IPv4 group 1 and IPv4 group 2 use private IPv4 addresses. Configure an IPv4 over IPv4 tunnel between Device A and Device B to make the two subnets reachable to each other.

Figure 14 Network diagram



Procedure

1. Configure Device A:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 24
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create IPv4 over IPv4 tunnel interface **Tunnel 1**.

```
[DeviceA] interface tunnel 1 mode ipv4-ipv4
[DeviceA-Tunnel1] ip address 10.1.2.1 255.255.255.0
[DeviceA-Tunnel1] source 2.1.1.1
[DeviceA-Tunnel1] destination 3.1.1.1
[DeviceA-Tunnel1] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 1 and the next hop is 2.1.1.2.

```
[DeviceA] ip route-static 3.1.1.1 24 2.1.1.2
[DeviceA] ip route-static 10.1.3.0 24 tunnel 1
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name tunnellocalout
[DeviceA-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-tunnellocalout] source-ip-host 2.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] destination-ip-host 3.1.1.1
[DeviceA-security-policy-ip-1-tunnellocalout] action pass
[DeviceA-security-policy-ip-1-tunnellocalout] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 group 1 to IPv4 group 2.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-2-trust-untrust] destination-ip-subnet 10.1.3.0 24
[DeviceA-security-policy-ip-2-trust-untrust] action pass
[DeviceA-security-policy-ip-2-trust-untrust] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name tunnellocalin
[DeviceA-security-policy-ip-3-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ip-3-tunnellocalin] destination-zone local
[DeviceA-security-policy-ip-3-tunnellocalin] source-ip-host 3.1.1.1
[DeviceA-security-policy-ip-3-tunnellocalin] destination-ip-host 2.1.1.1
[DeviceA-security-policy-ip-3-tunnellocalin] action pass
[DeviceA-security-policy-ip-3-tunnellocalin] quit
```

# Configure a rule named **untrust-trust** in the IPv4 security policy to permit the packets from IPv4 group 2 to IPv4 group 1.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-4-untrust-trust] source-ip-subnet 10.1.3.0 24
[DeviceA-security-policy-ip-4-untrust-trust] destination-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-4-untrust-trust] action pass
[DeviceA-security-policy-ip-4-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.3.1 24
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create IPv4 over IPv4 tunnel interface **Tunnel 2**.

```
[DeviceB] interface tunnel 2 mode ipv4-ipv4
[DeviceB-Tunnel2] ip address 10.1.2.2 255.255.255.0
[DeviceB-Tunnel2] source 3.1.1.1
[DeviceB-Tunnel2] destination 2.1.1.1
[DeviceB-Tunnel2] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 2 and the next hop is 3.1.1.2.

```
[DeviceB] ip route-static 2.1.1.1 24 3.1.1.2
[DeviceB] ip route-static 10.1.1.0 24 tunnel 2
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface tunnel 2
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
```

```
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name tunnellocalout
[DeviceB-security-policy-ip-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ip-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-tunnellocalout] source-ip-host 3.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] destination-ip-host 2.1.1.1
[DeviceB-security-policy-ip-1-tunnellocalout] action pass
[DeviceB-security-policy-ip-1-tunnellocalout] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 group 2 to IPv4 group 1.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 10.1.3.0 24
[DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-2-trust-untrust] action pass
[DeviceB-security-policy-ip-2-trust-untrust] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-3-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-3-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-3-tunnellocalin] source-ip-host 2.1.1.1
[DeviceB-security-policy-ip-3-tunnellocalin] destination-ip-host 3.1.1.1
[DeviceB-security-policy-ip-3-tunnellocalin] action pass
[DeviceB-security-policy-ip-3-tunnellocalin] quit
```

# Configure a rule named **tunnellocalin** in the IPv4 security policy to permit the packets from IPv4 group 1 to IPv4 group 2.

```
[DeviceB-security-policy-ip] rule name tunnellocalin
[DeviceB-security-policy-ip-4-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ip-4-tunnellocalin] destination-zone local
[DeviceB-security-policy-ip-4-tunnellocalin] source-ip-subnet 10.1.1.0 24
[DeviceB-security-policy-ip-4-tunnellocalin] destination-ip-subnet 10.1.3.0 24
[DeviceB-security-policy-ip-4-tunnellocalin] action pass
[DeviceB-security-policy-ip-4-tunnellocalin] quit
[DeviceB-security-policy-ip] quit
```

Verifying the configuration

# Use the **`display interface tunnel`** command to display the status of the tunnel interfaces on Device A and Device B. Verify that the tunnel interfaces are up. (Details not shown.)

# Verify that Device A and Device B can ping the IPv4 address of the peer interface GigabitEthernet 1/0/1. This example uses Device A.

```
[DeviceA] ping -a 10.1.1.1 10.1.3.1
Ping 10.1.3.1 (10.1.3.1) from 10.1.1.1: 56 data bytes, press CTRL_C to break
```

```
56 bytes from 10.1.3.1: icmp_seq=0 ttl=255 time=2.000 ms
56 bytes from 10.1.3.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.3.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.3.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.3.1: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 10.1.3.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.000/2.000/0.632 ms
```

IPv4 over IPv6 tunneling

About IPv4 over IPv6 tunneling

Implementation

IPv4 over IPv6 tunneling adds an IPv6 header to IPv4 packets so that the IPv4 packets can pass an IPv6 network through a tunnel to realize interworking between isolated IPv4 networks.

Figure 15 IPv4 over IPv6 tunnel

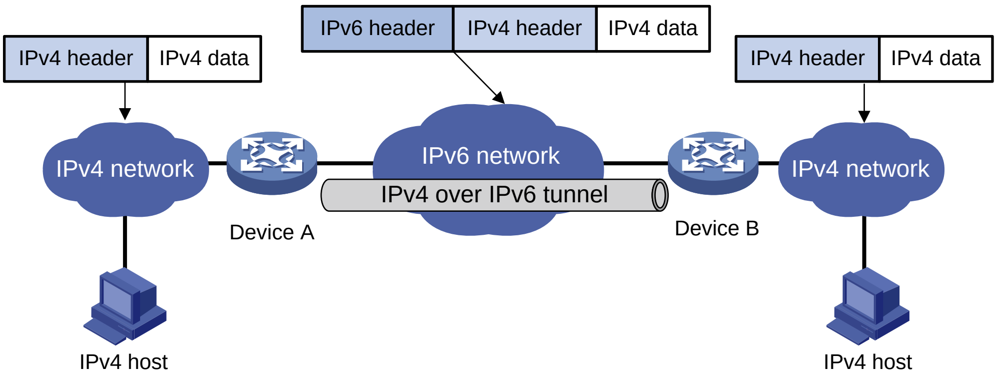


Figure 15 shows the encapsulation and de-encapsulation processes.

- Encapsulation:
  - **a.** Upon receiving an IPv4 packet, Device A delivers it to the IPv4 protocol stack.
  - **b.** The IPv4 protocol stack uses the destination address of the packet to determine the egress interface. If the egress interface is the tunnel interface, the IPv4 protocol stack delivers the packet to the tunnel interface.
  - **c.** The tunnel interface adds an IPv6 header to the original IPv4 packet and delivers the packet to the IPv6 protocol stack.
  - **d.** The IPv6 protocol stack uses the destination IPv6 address of the packet to look up the routing table, and then sends it out.
- De-encapsulation:
  - **a.** Upon receiving the IPv6 packet from the attached IPv6 network, Device B delivers the packet to the IPv6 protocol stack to examine the protocol type encapsulated in the data portion of the packet.
  - **b.** If the protocol type is IPv4, the IPv6 protocol stack delivers the packet to the tunneling module.
  - **c.** The tunneling module removes the IPv6 header and delivers the remaining IPv4 packet to the IPv4 protocol stack.
  - **d.** The IPv4 protocol stack forwards the IPv4 packet.

Tunnel modes

IPv4 over IPv6 tunnels include IPv4 over IPv6 manual tunnels and DS-Lite tunnels.

IPv4 over IPv6 manual tunnel

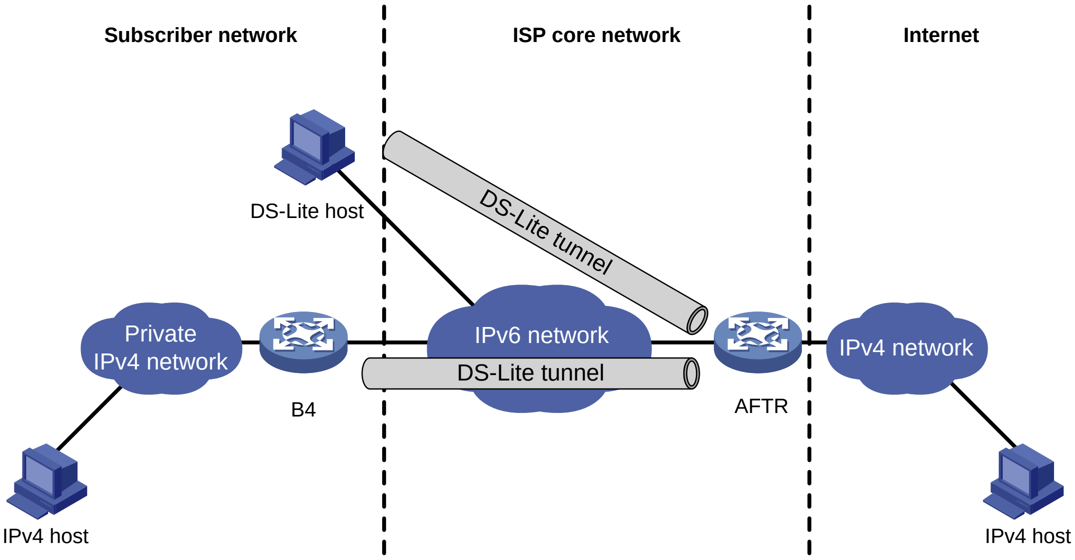
An IPv4 over IPv6 manual tunnel is a point-to-point link and its source and destination IPv6 addresses are manually configured. You can establish an IPv4 over IPv6 manual tunnel to connect isolated IPv4 networks over an IPv6 network.

DS-Lite tunnel

Dual Stack Lite (DS-Lite) is a combination of the tunneling and NAT technologies. NAT translates the private IPv4 addresses of the IPv4 hosts before the hosts reach the IPv4 public network.

DS-Lite tunnel supports only an IPv4 host in a private network initiating communication with an IPv4 host on the Internet. It does not support an IPv4 host on the Internet initiating communication with an IPv4 host in a private network.

Figure 16 DS-Lite tunnel



As shown in Figure 16, the DS-Lite feature contains the following components:

- Basic Bridging BroadBand (B4) element

  The B4 element is typically a CPE router that connects end hosts. IPv4 packets entering the B4 router are encapsulated into IPv6 packets and sent to the AFTR. IPv6 packets from the AFTR are de-encapsulated into IPv4 packets and sent to the subscriber's network.

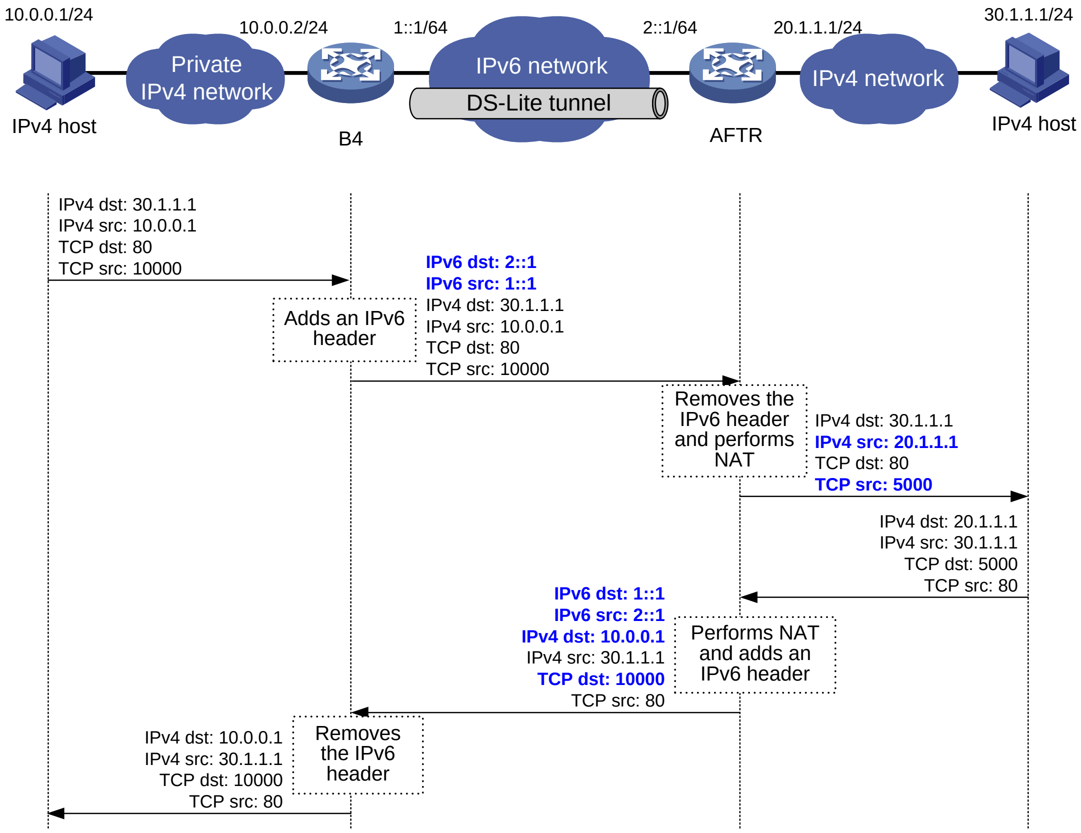
  Hosts that can act as the B4 router are referred to as DS-Lite hosts.

- Address Family Transition Router (AFTR)

  An AFTR resides in the ISP network and terminates the tunnel from the B4 router. NAT is also implemented on the interface that is connected to the public IPv4 network.

  An AFTR de-encapsulates the tunneled packet, translates the network address, and routes the packet to the destination IPv4 network. For IPv4 packets coming from the public IPv4 network, the AFTR performs reverse address translation and sends them to the B4 router by using the DS-Lite tunnel.

Figure 17 Packet forwarding process in DS-Lite



As shown in Figure 17, the packet forwarding process in DS-Lite is as follows:

1. Upon receiving a packet from the private IPv4 network, the B4 router adds an IPv6 header to the packet and sends the IPv6 packet to the AFTR through the tunnel.
2. The AFTR performs the following operations:
   a. Removes the IPv6 header from the tunneled packet.
   b. Assigns a tunnel ID for the B4 router.
   c. Records the mapping between the IPv6 address of the B4 router (the source IPv6 address of the packet), and the tunnel ID.
3. After de-encapsulation, the AFTR translates the source private IPv4 address of the packet into a public IPv4 address and sends the packet to the destination IPv4 host. The AFTR also maps the NAT entries to the tunnel ID so that IPv4 networks connected to different B4 routers can use the same address space.
4. Upon receiving the response packet from the public network, the AFTR translates the destination public IPv4 address into the private IPv4 address. The AFTR performs the following operations:
   a. Looks up the IPv6 address-tunnel ID mapping to obtain the IP address of the B4 router.
   b. Uses the address as the destination address of the encapsulated IPv6 packet.
   c. Forwards the packet to the B4 router.

Figure 17 shows an example of PAT translation for dynamic NAT. Typically, dynamic NAT is used. When you use static NAT for DS-Lite tunneling, make sure the IP addresses of private IPv4 networks connected to different B4 routers do not overlap. For more information about NAT, see *NAT Configuration Guide*.

Configuring an IPv4 over IPv6 manual tunnel

Restrictions and guidelines

When you perform the tasks in this section, follow these restrictions and guidelines:

- The tunnel destination address specified on the local device must be identical with the tunnel source address specified on the tunnel peer device.
- Do not specify the same tunnel source and destination addresses for tunnels of the same tunnel mode.
- To ensure correct packet forwarding, identify whether the destination IPv4 network and the IPv4 address of the local tunnel interface are on the same subnet. If they are not, configure a route reaching the destination IPv4 network through the tunnel interface. You can configure the route by using one of the following methods:
    - Configure a static route, and specify the local tunnel interface as the egress interface or specify the IPv6 address of the peer tunnel interface as the next hop.
    - Enable a dynamic routing protocol on both the local and remote tunnel interfaces.

  For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.

Procedure

1. Enter system view.
   `system-view`
2. Enter IPv6 tunnel interface view.
   `interface tunnel` *`number`* `[` **`mode ipv6`** `]`
3. Configure an IPv4 address for the tunnel interface.
   `ip address` *`ip-address`* `{` *`mask`* `|` *`mask-length`* `} [` **`sub`** `]`
4. Configure a source address or interface for the tunnel.
   `source` `{` *`ipv6-address`* `|` *`interface-type interface-number`* `}`

   By default, no source address or interface is configured for a tunnel.

   If you specify a source address, it is used as the source IPv6 address of tunneled packets.

   If you specify a source interface, the lowest IPv6 address of this interface is used as the source IPv6 address of tunneled packets.
5. Configure a destination address for the tunnel.
   `destination` *`ipv6-address`*

   By default, no destination address is configured for a tunnel.

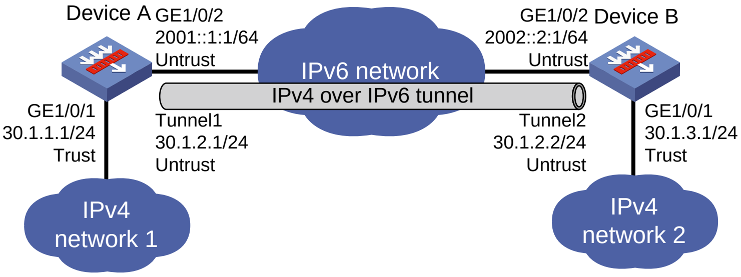
   The tunnel destination address must be the IPv6 address of the receiving interface on the tunnel peer. It is used as the destination IPv6 address of tunneled packets.

Example: Configuring an IPv4 over IPv6 manual tunnel

Network configuration

As shown in Figure 18, configure an IPv4 over IPv6 manual tunnel between Device A and Device B so the two networks can reach each other over the IPv6 network.

Figure 18 Network diagram



Procedure

1. Configure Device A:

   # Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 30.1.1.1 24
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipv6 address 2001::1:1 64
[DeviceA-GigabitEthernet1/0/2] quit
```

   # Create IPv6 tunnel interface **Tunnel 1**.

```
[DeviceA] interface tunnel 1 mode ipv6
[DeviceA-Tunnel1] ip address 30.1.2.1 255.255.255.0
[DeviceA-Tunnel1] source 2001::1:1
[DeviceA-Tunnel1] destination 2002::2:1
[DeviceA-Tunnel1] quit
```

   # Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 1 and the next hop is 2001::1:2.

```
[DeviceA] ipv6 route-static 2002::2:1 64 2001::1:2
[DeviceA] ip route-static 30.1.3.0 24 tunnel 1
```

   # Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

   # Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalout
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-ip-host 2001::1:1
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-ip-host 2002::2:1
[DeviceA-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceA-security-policy-ipv6-1-tunnellocalout] quit
```

```
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 network 1 to IPv4 network 2.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] destination-ip-subnet 30.1.3.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalin
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-ip-host 2002::2:1
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-ip-host 2001::1:1
[DeviceA-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceA-security-policy-ipv6-2-tunnellocalin] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **untrust-trust** in the IPv4 security policy to permit the packets from IPv4 network 2 to IPv4 network 1.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-2-untrust-trust] source-ip-subnet 30.1.3.0 24
[DeviceA-security-policy-ip-2-untrust-trust] destination-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] action pass
[DeviceA-security-policy-ip-2-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 30.1.3.1 24
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipv6 address 2002::2:1 64
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create IPv6 tunnel interface **Tunnel 2**.

```
[DeviceB] interface tunnel 2 mode ipv6
[DeviceB-Tunnel2] ip address 30.1.2.2 255.255.255.0
[DeviceB-Tunnel2] source 2002::2:1
[DeviceB-Tunnel2] destination 2001::1:1
```

```
[DeviceB-Tunnel2] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 2 and the next hop is 2002::2:2.

```
[DeviceB] ipv6 route-static 2001::1:1 64 2002::2:2
[DeviceB] ip route-static 30.1.1.0 24 tunnel 2
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface tunnel 2
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalout
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-ip-host 2002::2:1
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-ip-host 2001::1:1
[DeviceB-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceB-security-policy-ipv6-1-tunnellocalout] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 network 2 to IPv4 network 1.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 30.1.3.0 24
[DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 30.1.1.0 24
[DeviceB-security-policy-ip-2-trust-untrust] action pass
[DeviceB-security-policy-ip-2-trust-untrust] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-ip-host 2.1.1.1
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-ip-host 3.1.1.1
[DeviceB-security-policy-ipv6-3-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-3-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **untrust-trust** in the IPv4 security policy to permit the packets from IPv4 network 1 to IPv4 network 2.

```
[DeviceB] security-policy ip
```

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 30.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 30.1.3.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

Verifying the configuration

# Use the **`display interface tunnel`** command to display the status of the tunnel interfaces on Device A and Device B. Verify that the tunnel interfaces are up. (Details not shown.)

# Verify that Device A and Device B can ping the IPv4 address of the peer interface GigabitEthernet 1/0/1. This example uses Device A.

```
[DeviceA] ping -a 30.1.1.1 30.1.3.1
Ping 30.1.3.1 (30.1.3.1) from 30.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 30.1.3.1: icmp_seq=0 ttl=255 time=3.000 ms
56 bytes from 30.1.3.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 30.1.3.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 30.1.3.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 30.1.3.1: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 30.1.3.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.200/3.000/0.980 ms
```

Configuring a DS-Lite tunnel

Restrictions and guidelines for DS-Lite tunnel configuration

A B4 tunnel interface can establish a tunnel with only one AFTR tunnel interface, but an AFTR tunnel interface can establish tunnels with multiple B4 tunnel interfaces.

Follow these guidelines when you configure the B4 router of a DS-Lite tunnel:

- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.
- The tunnel destination address specified on the B4 router must be the tunnel source address specified on the AFTR.
- To ensure correct packet forwarding, identify whether the destination IPv4 network and the IPv4 address of the local tunnel interface are on the same subnet. If they are not, configure a route reaching the destination IPv4 network through the tunnel interface. You can configure the route by using one of the following methods:
    - Configure a static route, and specify the local tunnel interface as the egress interface or specify the IPv6 address of the peer tunnel interface as the next hop.
    - Enable a dynamic routing protocol on both the local and remote tunnel interfaces.

  For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.
- For a B4 router to automatically establish a DS-Lite tunnel with an AFTR, configure DHCPv6 client, static or dynamic IPv6 DNS, and the **`destination dhcp-alloc`** command on the B4 router. In addition, make sure a DHCPv6 server and an IPv6 DNS server (for dynamic DNS) exist in the network. For more information about DHCPv6 server, DHCPv6 client, and IPv6 DNS, see *Layer 3—IP Services Configuration Guide*.

Follow these guidelines when you configure the AFTR of a DS-Lite tunnel:

- Do not specify the same tunnel source addresses for tunnels of the same tunnel mode.
- Enable NAT on the interface that is connected to the public IPv4 interface.
- The tunnel destination cannot be configured on the AFTR. The AFTR uses the address of the B4 router as the IPv6 address of the tunnel destination.
- It is not necessary to configure a route to the destination IPv4 address for forwarding packets through the tunnel interface.

Configuring the B4 router of a DS-Lite tunnel

1. Enter system view.

   system-view

2. Enter IPv6 tunnel interface view.

   interface tunnel *number* [**mode ipv6**]

3. Specify an IPv4 address for the tunnel interface.

   ip address *ip-address* { *mask* | *mask-length* } [**sub**]

4. Specify a source address or source interface for the tunnel.

   source { *ipv6-address* | *interface-type interface-number* }

   By default, no source address or interface is specified for a tunnel.

   If you specify a source address, it is used as the source IPv6 address of tunneled packets.

   If you specify a source interface, the lowest IPv6 address of this interface is used as the source IPv6 address of tunneled packets.

5. Specify a destination address for the tunnel.

   destination { *ipv6-address* | **dhcp-alloc** *interface-type interface-number* }

   By default, no destination address is configured for a tunnel.

   The tunnel destination address must be the IPv6 address of the receiving interface on the tunnel peer. It is used as the destination IPv6 address of tunneled packets.

Parameter	Remarks
ipv6-address	Specifies the tunnel source address configured on the AFTR as the destination address.
dhcp-alloc *interface-type interface-number*	Specifies an interface to obtain the AFTR's IPv6 address from the received DHCPv6 packets for automatic DS-Lite tunnel establishment.

Configuring the AFTR of a DS-Lite tunnel

1. Enter system view.

   system-view

2. Enter the view of the tunnel interface on the AFTR.

   interface tunnel *number* [**mode ds-lite-aftr**]

3. Specify an IPv4 address for the tunnel interface.

   ip address *ip-address* { *mask* | *mask-length* } [**sub**]

4. Specify a source address or source interface for the tunnel.

   source { *ipv6-address* | *interface-type interface-number* }

By default, no source address or interface is specified for a tunnel.

The specified source address or the lowest IPv6 address of the specified source interface is used as the source IPv6 address of tunneled packets.

5. Return to system view.

   quit

6. Enter the view of the interface that is connected to the IPv4 public network.

   interface *interface-type interface-number*

7. Enable DS-Lite tunneling on the interface.

   ds-lite enable

   By default, DS-Lite tunneling is disabled.

   Only after you use this command, the AFTR can tunnel IPv4 packets from the public IPv4 network to the B4 router.

8. Display information about the connected B4 routers on the AFTR.

   display ds-lite b4 information

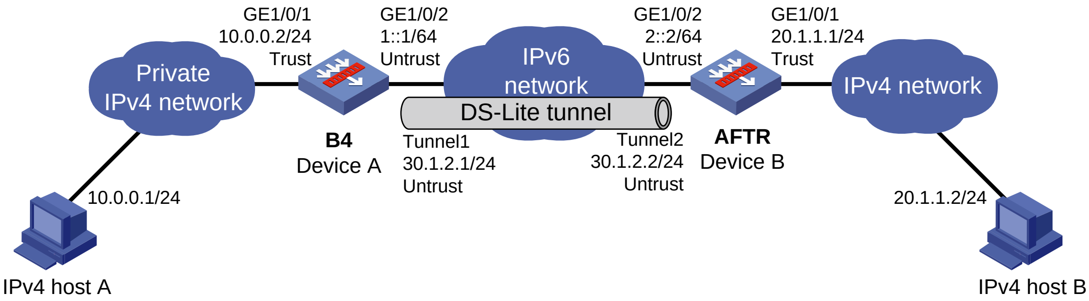
Example: Configuring a DS-Lite tunnel

Network configuration

As shown in Figure 19, to enable hosts in the private IPv4 network to access the public IPv4 network over the IPv6 network, perform the following tasks:

- Configure a DS-Lite tunnel between Device A and Device B.
- Configure NAT on GigabitEthernet 1/0/1 on the AFTR.

Figure 19 Network diagram



Procedure

1. Configure Device A:

   # Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.0.0.2 24
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipv6 address 1::1 64
[DeviceA-GigabitEthernet1/0/2] quit
```

   # Create IPv6 tunnel interface **Tunnel 1**.

```
[DeviceA] interface tunnel 1 mode ipv6
[DeviceA-Tunnel1] ip address 30.1.2.1 255.255.255.0
[DeviceA-Tunnel1] source 1::1
```

```
[DeviceA-Tunnel1] destination 2::2
[DeviceA-Tunnel1] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 1 and the next hop is 1::2.

```
[DeviceA] ipv6 route-static 2002::2:1 64 1::2
[DeviceA] ip route-static 20.1.1.0 24 tunnel 1
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalout
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-ip-host 1::1
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-ip-host 2::2
[DeviceA-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceA-security-policy-ipv6-1-tunnellocalout] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 host A to IPv4 host B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 10.0.0.0 24
[DeviceA-security-policy-ip-1-trust-untrust] destination-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalin
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-tunnellocalin] source-ip-host 2::2
[DeviceA-security-policy-ipv6-2-tunnellocalin] destination-ip-host 1::1
[DeviceA-security-policy-ipv6-2-tunnellocalin] action pass
[DeviceA-security-policy-ipv6-2-tunnellocalin] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **untrust-trust** in the IPv4 security policy to permit the packets from IPv4 host B to IPv4 host A.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-2-untrust-trust] source-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] destination-ip-subnet 10.0.0.0 24
[DeviceA-security-policy-ip-2-untrust-trust] action pass
[DeviceA-security-policy-ip-2-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 20.1.1.1 24
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipv6 address 2::2 64
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create DS-Lite tunnel interface **Tunnel 2**.

```
[DeviceB] interface tunnel 2 mode ds-lite-aftr
[DeviceB-Tunnel2] ip address 30.1.2.2 255.255.255.0
[DeviceB-Tunnel2] source gigabitethernet 1/0/2
[DeviceB-Tunnel2] quit
```

# Enable DS-Lite tunneling on GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ds-lite enable
```

# Enable NAT on GigabitEthernet 1/0/1 and use the IP address of GigabitEthernet 1/0/1 as the translated address.

```
[DeviceB-GigabitEthernet1/0/1] nat outbound
[DeviceB-GigabitEthernet1/0/1] quit
```

# Configure settings for routing. This example configures a static route, and the next hop in the route is 2::3.

```
[DeviceB] ipv6 route-static 1::1 64 2::3
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface tunnel 2
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalout
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-ip-host 2::2
```

```
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-ip-host 1::1
[DeviceB-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceB-security-policy-ipv6-1-tunnellocalout] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **trust-untrust** in the IPv4 security policy to permit the packets from IPv4 host B to IPv4 host A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 20.1.1.0 24
[DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 10.0.0.0 24
[DeviceB-security-policy-ip-2-trust-untrust] action pass
[DeviceB-security-policy-ip-2-trust-untrust] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-ip-host 1::1
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-ip-host 2::2
[DeviceB-security-policy-ipv6-3-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-3-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **untrust-trust** in the IPv4 security policy to permit the packets from IPv4 host A to IPv4 host B.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-4-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-4-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-4-untrust-trust] source-ip-subnet 10.0.0.0 24
[DeviceB-security-policy-ip-4-untrust-trust] destination-ip-subnet 20.1.1.0 24
[DeviceB-security-policy-ip-4-untrust-trust] action pass
[DeviceB-security-policy-ip-4-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

3. On host A, specify the IP address for the host as 10.0.0.1 and configure a static route to 20.1.1.0/24 with next hop 10.0.0.2. (Details not shown.)
4. On host B, specify the IP address for the host as 20.1.1.2. (Details not shown.)

Verifying the configuration

# Use the **`display interface tunnel`** command to display the status of the tunnel interfaces on Device A and Device B. Verify that the tunnel interfaces are up. (Details not shown.)

# Verify that host A can ping host B.

```
C:\> ping 20.1.1.2
Pinging 20.1.1.2 with 32 bytes of data:
Reply from 20.1.1.2: bytes=32 time=51ms TTL=255
Reply from 20.1.1.2: bytes=32 time=44ms TTL=255
Reply from 20.1.1.2: bytes=32 time=1ms TTL=255
Reply from 20.1.1.2: bytes=32 time=1ms TTL=255
Ping statistics for 20.1.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

IPv6 over IPv6 tunneling

About IPv6 over IPv6 tunneling

IPv6 over IPv6 tunneling (RFC 2473) enables isolated IPv6 networks to communicate with each other over another IPv6 network. For example, two isolated IPv6 networks that do not want to show their addresses to the Internet can use an IPv6 over IPv6 tunnel to communicate with each other.

Figure 20 Principle of IPv6 over IPv6 tunneling

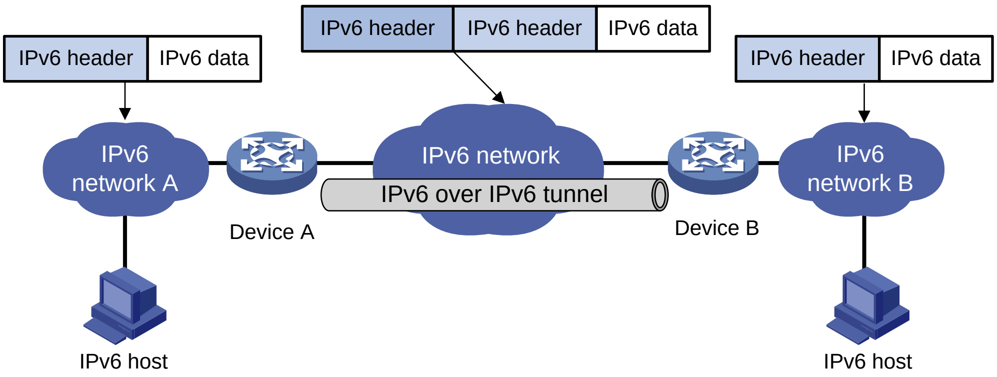


Figure 20 shows the encapsulation and de-encapsulation processes.

- Encapsulation:
  - **a.** After receiving an IPv6 packet, Device A submits it to the IPv6 protocol stack.
  - **b.** The IPv6 protocol stack uses the destination IPv6 address of the packet to find the egress interface. If the egress interface is the tunnel interface, the stack delivers it to the tunnel interface.
  - **c.** After receiving the packet, the tunnel interface adds an IPv6 header to it and submits it to the IPv6 protocol stack.
  - **d.** The IPv6 protocol stack forwards the packet according to its destination IPv6 address.
- De-encapsulation:
  - **a.** Upon receiving the IPv6 packet, Device B delivers it to the IPv6 protocol stack.
  - **b.** The IPv6 protocol stack checks the protocol type of the data portion encapsulated in the IPv6 packet. If the encapsulation protocol is IPv6, the stack delivers the packet to the tunnel module.
  - **c.** The tunnel module de-encapsulates the packet and sends it back to the IPv6 protocol stack.
  - **d.** The IPv6 protocol stack forwards the IPv6 packet.

Restrictions and guidelines: IPv6 over IPv6 tunnel configuration

Follow these guidelines when you configure an IPv6 over IPv6 tunnel:

- The tunnel destination address specified on the local device must be identical with the tunnel source address specified on the tunnel peer device.
- Do not specify the same tunnel source and destination addresses for tunnels of the same tunnel mode.

- The IPv6 address of the tunnel interface cannot be on the same subnet as the destination address configured for the tunnel interface.
- To ensure correct packet forwarding, identify whether the destination IPv6 network and the IPv6 address of the local tunnel interface are on the same subnet. If they are not, configure a route reaching the destination IPv6 network through the tunnel interface. You can configure the route by using one of the following methods:
  - Configure a static route, and specify the local tunnel interface as the egress interface or specify the IPv6 address of the peer tunnel interface as the next hop.
  - Enable a dynamic routing protocol on both the local and remote tunnel interfaces.

  For more information about route configuration, see *Layer 3—IP Routing Configuration Guide*.
- The destination address of the route passing the tunnel interface cannot be on the same subnet as the destination address configured on the tunnel interface.

IPv6 over IPv6 tunnel configuration tasks at a glance

To configure an IPv6 over IPv6 tunnel, perform the following tasks:

1. Configuring an IPv6 over IPv6 tunnel
2. (Optional.) Enabling dropping IPv6 packets that use IPv4-compatible IPv6 addresses

Configuring an IPv6 over IPv6 tunnel

1. Enter system view.

   system-view
2. Enter IPv6 tunnel interface view.

   interface tunnel *number* [**mode ipv6**]
3. Configure an IPv6 address for the tunnel interface.

   See "Configuring basic IPv6 settings."
4. Configure a source address or source interface for the tunnel.

   source { *ipv6-address* | *interface-type interface-number* }

   By default, no source address or interface is configured for a tunnel.

   If you specify a source address, it is used as the source IPv6 address of tunneled packets.

   If you specify a source interface, the lowest IPv6 address of this interface is used as the source IPv6 address of tunneled packets.
5. Configure a destination address for the tunnel.

   destination *ipv6-address*

   By default, no destination address is configured for a tunnel.

   The tunnel destination address must be the IPv6 address of the receiving interface on the tunnel peer. It is used as the destination IPv6 address of tunneled packets.
6. (Optional.) Set the maximum number of nested encapsulations of a packet.

   encapsulation-limit *number*

   By default, there is no limit to the nested encapsulations of a packet.

Enabling dropping IPv6 packets that use IPv4-compatible IPv6 addresses

1. Enter system view.

   `system-view`

2. Enable dropping IPv6 packets that use IPv4-compatible IPv6 addresses.

   `tunnel discard ipv4-compatible-packet`

   By default, IPv6 packets that use IPv4-compatible IPv6 addresses are not dropped.

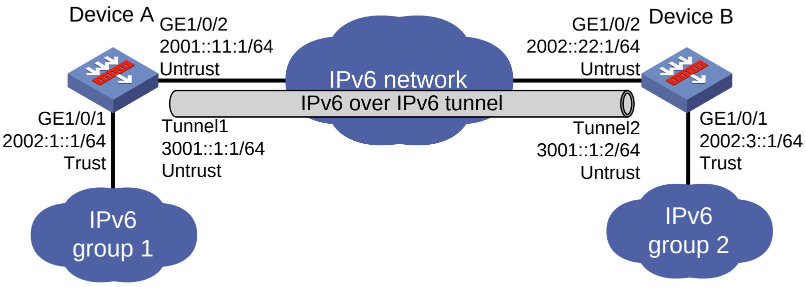
IPv6 over IPv6 tunnel configuration examples

Example: Configuring an IPv6 over IPv6 tunnel

Network configuration

As shown in Figure 21, configure an IPv6 over IPv6 tunnel between Device A and Device B so the two networks can reach each other without disclosing their IPv6 addresses.

Figure 21 Network diagram



Procedure

1. Configure Device A:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ipv6 address 2002:1::1 64
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   # Create IPv6 tunnel interface **Tunnel 1**.

   ```
   [DeviceA] interface tunnel 1 mode ipv6
   [DeviceA-Tunnel1] ipv6 address 3001::1:1 64
   [DeviceA-Tunnel1] source 2001::11:1
   [DeviceA-Tunnel1] destination 2002::22:1
   [DeviceA-Tunnel1] quit
   ```

   # Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 1 and the next hop is 2001::11:2.

   ```
   [DeviceA] ipv6 route-static 2002::22:1 64 2001::11:2
   [DeviceA] ipv6 route-static 2002:3:: 64 tunnel 1
   ```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface tunnel 1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name tunnellocalout
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-tunnellocalout] source-ip-host 2001::11:1
[DeviceA-security-policy-ipv6-1-tunnellocalout] destination-ip-host 2002::22:1
[DeviceA-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceA-security-policy-ipv6-1-tunnellocalout] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from IPv6 group 1 to IPv6 group 2.

```
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-2-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-2-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-2-trust-untrust] source-ip-subnet 2002:1::1 64
[DeviceA-security-policy-ipv6-2-trust-untrust] destination-ip-subnet 2002:3::1 64
[DeviceA-security-policy-ipv6-2-trust-untrust] action pass
[DeviceA-security-policy-ipv6-2-trust-untrust] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ipv6] rule name tunnellocalin
[DeviceA-security-policy-ipv6-3-tunnellocalin] source-zone untrust
[DeviceA-security-policy-ipv6-3-tunnellocalin] destination-zone local
[DeviceA-security-policy-ipv6-3-tunnellocalin] source-ip-host 2002::22:1
[DeviceA-security-policy-ipv6-3-tunnellocalin] destination-ip-host 2001::11:1
[DeviceA-security-policy-ipv6-3-tunnellocalin] action pass
[DeviceA-security-policy-ipv6-3-tunnellocalin] quit
```

# Configure a rule named **untrust-trust** in the IPv6 security policy to permit the packets from IPv6 group 2 to IPv6 group 1.

```
[DeviceA-security-policy-ipv6] rule name untrust-trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-zone untrust
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-zone trust
[DeviceA-security-policy-ipv6-4-untrust-trust] source-ip-subnet 2002:3::1 64
[DeviceA-security-policy-ipv6-4-untrust-trust] destination-ip-subnet 2002:1::1 64
[DeviceA-security-policy-ipv6-4-untrust-trust] action pass
[DeviceA-security-policy-ipv6-4-untrust-trust] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
```

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2002:3::1 64
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create IPv6 tunnel interface **Tunnel 2**.

```
[DeviceB] interface tunnel 2 mode ipv6
[DeviceB-Tunnel2] ipv6 address 3001::1:2 64
[DeviceB-Tunnel2] source 2002::22:1
[DeviceB-Tunnel2] destination 2001::11:1
[DeviceB-Tunnel2] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 2 and the next hop is 2002::22:2.

```
[DeviceB] ipv6 route-static 2001::1:1 64 2002::22:2
[DeviceB] ipv6 route-static 2002:1::1 64 tunnel 2
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface tunnel 2
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **tunnellocalout** in the IPv6 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name tunnellocalout
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-zone local
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-tunnellocalout] source-ip-host 2002::22:1
[DeviceB-security-policy-ipv6-1-tunnellocalout] destination-ip-host 2001::11:1
[DeviceB-security-policy-ipv6-1-tunnellocalout] action pass
[DeviceB-security-policy-ipv6-1-tunnellocalout] quit
```

# Configure a rule named **trust-untrust** in the IPv6 security policy to permit the packets from IPv6 group 2 to IPv6 group 1.

```
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-2-trust-untrust] source-ip-subnet 2002:3::1 64
[DeviceB-security-policy-ipv6-2-trust-untrust] destination-ip-subnet 2002:1::1 64
[DeviceB-security-policy-ipv6-2-trust-untrust] action pass
[DeviceB-security-policy-ipv6-2-trust-untrust] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-3-tunnellocalin] source-ip-host 2001::11:1
[DeviceB-security-policy-ipv6-3-tunnellocalin] destination-ip-host 2002::22:1
```

```
[DeviceB-security-policy-ipv6-3-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-3-tunnellocalin] quit
```

# Configure a rule named **tunnellocalin** in the IPv6 security policy to permit the packets from IPv6 group 1 to IPv6 group 2.

```
[DeviceB-security-policy-ipv6] rule name tunnellocalin
[DeviceB-security-policy-ipv6-4-tunnellocalin] source-zone untrust
[DeviceB-security-policy-ipv6-4-tunnellocalin] destination-zone local
[DeviceB-security-policy-ipv6-4-tunnellocalin] source-ip-subnet 2002:1::1 64
[DeviceB-security-policy-ipv6-4-tunnellocalin] destination-ip-subnet 2002:3::1 64
[DeviceB-security-policy-ipv6-4-tunnellocalin] action pass
[DeviceB-security-policy-ipv6-4-tunnellocalin] quit
[DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Use the **`display ipv6 interface`** command to display the status of the tunnel interfaces on Device A and Device B. Verify that the tunnel interfaces are up. (Details not shown.)

# Verify that Device A and Device B can ping the IPv6 address of the peer interface GigabitEthernet 1/0/1. This example uses Device A.

```
[DeviceA] ping ipv6 -a 2002:1::1 2002:3::1
Ping6(56 data bytes) 2002:1::1 --> 2002:3::1, press CTRL_C to break
56 bytes from 2002:3::1, icmp_seq=0 hlim=64 time=9.000 ms
56 bytes from 2002:3::1, icmp_seq=1 hlim=64 time=1.000 ms
56 bytes from 2002:3::1, icmp_seq=2 hlim=64 time=0.000 ms
56 bytes from 2002:3::1, icmp_seq=3 hlim=64 time=0.000 ms
56 bytes from 2002:3::1, icmp_seq=4 hlim=64 time=0.000 ms

--- Ping6 statistics for 2002:3::1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/2.000/9.000/3.521 ms
```

Contents

Configuring GRE

About GRE

Generic Routing Encapsulation (GRE) is a tunneling protocol that can encapsulate a protocol (such as IP or Ethernet) into a virtual point-to-point tunnel over a network (such as an IP network). Packets are encapsulated at one tunnel end and de-encapsulated at the other tunnel end. The network layer protocol of the packets before encapsulation and after encapsulation can be the same or different.

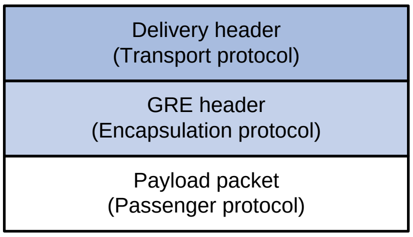
GRE encapsulation format

As shown in Figure 1, a GRE-tunneled packet includes the following parts:

- **Payload packet**—Original packet. The protocol type of the payload packet is called the passenger protocol. The passenger protocol can be any network layer protocol.
- **GRE header**—Header that is added to the payload packet to change the payload packet to a GRE packet. A GRE header includes the number of encapsulations, version, passenger protocol type, checksum, and key. GRE is called the encapsulation protocol.
- **Delivery header**—Header that is added to the GRE packet to deliver it to the tunnel end. The transport protocol (or delivery protocol) is the network layer protocol that transfers GRE packets.

The device supports GRE tunnels with IPv4 and IPv6 as the transport protocols. When the transport protocol is IPv4, the GRE tunnel mode is GRE over IPv4 (GRE/IPv4). When the transport protocol is IPv6, the GRE tunnel mode is GRE over IPv6 (GRE/IPv6).

Figure 1 GRE encapsulation format



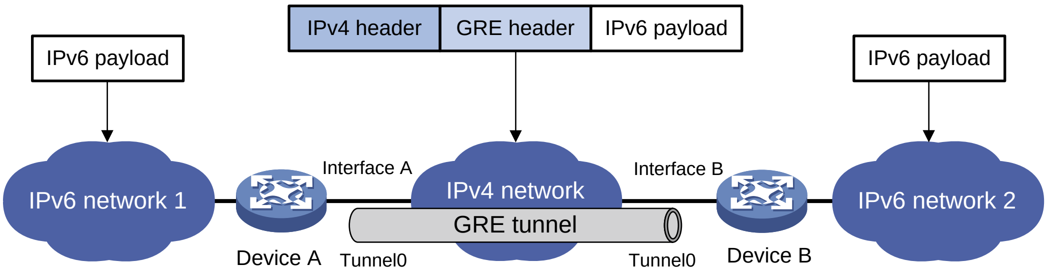
GRE tunnel operating principle

As shown in Figure 2, an IPv6 protocol packet traverses an IPv4 network through a GRE tunnel as follows:

1. After receiving an IPv6 packet from the interface connected to IPv6 network 1, Device A processes the packet as follows:
   a. Looks up the routing table to identify the outgoing interface for the IPv6 packet.
   b. Submits the IPv6 packet to the outgoing interface—the GRE tunnel interface Tunnel 0.
2. Upon receiving the packet, the tunnel interface encapsulates the packet with GRE and then with IPv4. In the IPv4 header:
   - The source address is the tunnel's source address (the IP address of Interface A of Device A).
   - The destination address is the tunnel's destination address (the IP address of Interface B of Device B).

3. Device A looks up the routing table according to the destination address in the IPv4 header, and forwards the IPv4 packet out of the physical interface (Interface A) of the GRE tunnel.
4. When the IPv4 arrives at the GRE tunnel destination Device B, Device B checks the destination address. Because the destination is Device B itself and the protocol number in the IP header is 47 (the protocol number for GRE), Device B submits the packet to GRE for de-encapsulation.
5. GRE first removes the IPv4 header, and then checks the GRE key, checksum, and packet sequence number. After GRE finishes the checking, it removes the GRE header, and submits the payload to the IPv6 protocol for forwarding.

Figure 2 IPv6 networks interconnected through a GRE tunnel



GRE security mechanisms

GRE supports the GRE key and GRE checksum security mechanisms.

GRE key

GRE keys ensure packet validity. The sender adds a GRE key into a packet. The receiver compares the GRE key with its own GRE key. If the two keys are the same, the receiver accepts the packet. If the two keys are different, the receiver drops the packet.

GRE checksum

GRE checksums ensure packet integrity. The sender calculates a checksum for the GRE header and payload and sends the packet containing the checksum to the tunnel peer. The receiver calculates a checksum for the received packet and compares it with that carried in the packet. If the checksums are the same, the receiver considers the packet intact and continues to process the packet. If the checksums are different, the receiver discards the packet.

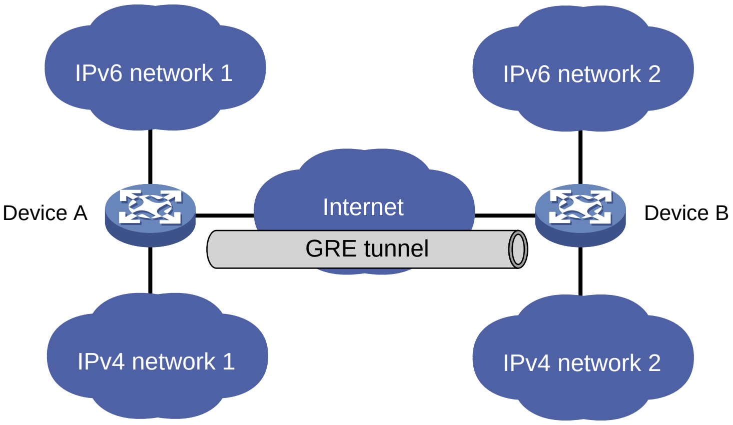
GRE application scenarios

The following shows typical GRE application scenarios:

Connecting networks running different protocols over a single backbone

As shown in Figure 3, IPv6 network 1 and IPv6 network 2 are IPv6 networks, and IPv4 network 1 and IPv4 network 2 are IPv4 networks. Through the GRE tunnel between Device A and Device B, IPv6 network 1 can communicate with IPv6 network 2 and IPv4 network 1 can communicate with IPv4 network 2, without affecting each other.

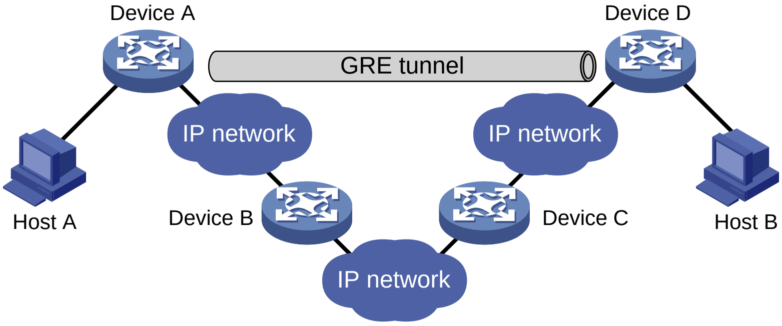
Figure 3 Network diagram



Enlarging network scope

In an IP network, the maximum TTL value of a packet is 255. If two devices have more than 255 hops in between, they cannot communicate with each other. By using a GRE tunnel, you can hide some hops to enlarge the network scope. As shown in Figure 4, only the tunnel-end devices (Device A and Device D) of the GRE tunnel are counted in hop count calculation. Therefore, there are only three hops between Host A and Host B.

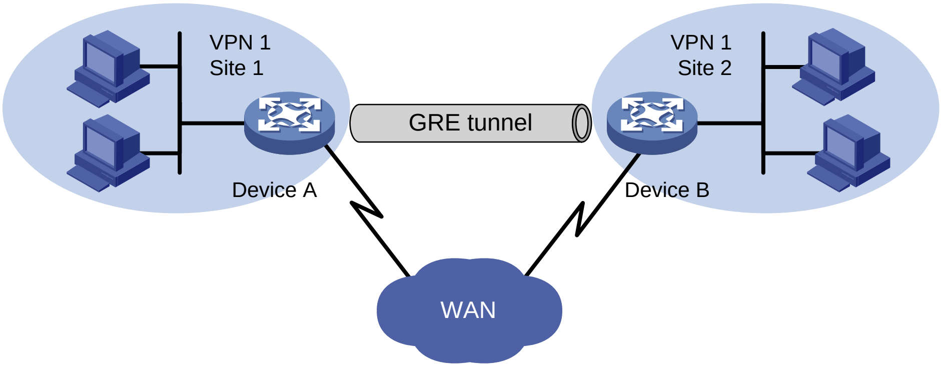
Figure 4 Network diagram



Constructing VPN

As shown in Figure 5, Site 1 and Site 2 both belong to VPN 1 and are located in different cities. Using a GRE tunnel can connect the two VPN sites across the WAN.

Figure 5 Network diagram

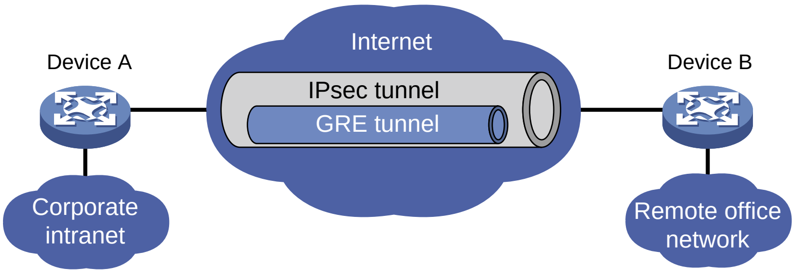


Operating with IPsec

As shown in Figure 6, GRE can be used together with IPsec to form a GRE over IPsec tunnel. Packets (for example, routing protocol packets, voice data, and video data) are first encapsulated with GRE and then with IPsec. GRE over IPsec delivers the following benefits:

- Improves transmission security.
- Allows IPsec to protect not only unicast packets. GRE supports encapsulating multicast, broadcast, and non-IP packets. After GRE encapsulation, these packets become common unicast packets, which can be protected by IPsec.
- Simplifies IPsec configuration. Packets are first encapsulated by GRE. You can define the packets to be protected by IPsec according to the GRE tunnel's source and destination addresses, without considering the source and destination addresses of the original packets.

Figure 6 Network diagram



GRE and IPsec can also form IPsec over GRE tunnels. As a best practice, use GRE over IPsec tunnels instead of IPsec over GRE tunnels.

For more information about IPsec, see *Security Configuration Guide*.

Protocols and standards

- RFC 1701, *Generic Routing Encapsulation (GRE)*
- RFC 1702, *Generic Routing Encapsulation over IPv4 networks*
- RFC 2784, *Generic Routing Encapsulation (GRE)*
- RFC 2890, *Key and Sequence Number Extensions to GRE*

Restrictions and guidelines: GRE configuration

When you configure a GRE tunnel, follow these restrictions and guidelines:

- You must configure the tunnel source address and destination address at both ends of a tunnel. The tunnel source or destination address at one end must be the tunnel destination or source address at the other end.
- As a best practice, do not configure the same tunnel source and destination addresses for local tunnel interfaces that use the same tunnel mode.
- Do not configure the same tunnel source and destination addresses for a GRE tunnel interface and a GRE P2MP tunnel interface.
- You can enable or disable GRE checksum at each end of a tunnel. If GRE checksum is enabled at a tunnel end, the tunnel end sends packets carrying the checksum to the peer end. A tunnel end checks the GRE checksum of a received packet if the packet carries a GRE checksum, whether or not the tunnel end is enabled with GRE checksum.
- To ensure correct packet forwarding, identify whether the destination network of packets and the IP address of the local tunnel interface are on the same subnet. If they are not, configure a

route reaching the destination network through the tunnel interface. You can configure the route by using one of the following methods:

  - Configure a static route, using the local tunnel interface as the outgoing interface of the route.
  - Enable a dynamic routing protocol on both the tunnel interface and the interface connecting the private network. This allows the dynamic routing protocol to establish a routing entry with the tunnel interface as the outgoing interface.

- GRE encapsulation and de-encapsulation can decrease the forwarding efficiency of tunnel-end devices.

Configuring a GRE/IPv4 tunnel

Restrictions and guidelines

This task describes only GRE/IPv4 tunnel required tunnel interface commands (the **`interface tunnel`**, **`source`**, and **`destination`** commands). For more tunnel interface commands, see "Configuring tunneling."

Procedure

1. Enter system view.

   `system-view`

2. Create a GRE tunnel interface, and specify the tunnel mode as GRE/IPv4.

   `interface tunnel` *`number`* **`mode gre`**

   You must configure the same tunnel mode on both ends of a tunnel. Otherwise, packet delivery might fail.

3. Configure an IP address for the tunnel interface based on the passenger protocol.

   IPv4:

   For information about how to assign an IPv4 address to an interface, see "Configuring IP addressing."

   IPv6:

   For information about how to assign an IPv6 address to an interface, see "Configuring basic IPv6 settings."

   By default, no IP address is configured for a tunnel interface.

4. Configure a source address or source interface for the tunnel.

   `source` { *`ip-address`* | *`interface-type interface-number`* }

   By default, no source address or interface is configured for a tunnel.

   If you configure a source address, the tunnel interface uses the source address as the source address of encapsulated packets.

   If you configure a source interface, the tunnel interface uses the primary IP address of the source interface as the source address of encapsulated packets.

5. Configure a destination address for the tunnel.

   `destination` *`ip-address`*

   By default, no destination address is configured for a tunnel.

   The destination address is the address of the physical interface that the tunnel remote end uses to receive packets from the GRE tunnel.

   The tunnel local end uses this address as the destination address of encapsulated packets.

   The tunnel destination address and the IP address of the tunnel interface must be in different subnets.

6. (Optional.) Enable GRE keepalive, and set the keepalive interval and keepalive number.

keepalive [*interval* [*times*]]

By default, GRE keepalive is disabled.

7. (Optional.) Configure GRE security mechanisms.
   - Enable GRE checksum.

     gre checksum

     By default, GRE checksum is disabled.
   - Configure a GRE key for the GRE tunnel interface.

     gre key *key*

     By default, no GRE key is configured for a GRE tunnel interface.

     The two ends of a GRE tunnel must have the same key or both have no key.
8. (Optional.) Set the DF bit for encapsulated packets.

   tunnel dfbit enable

   By default, the DF bit is not set, allowing encapsulated packets to be fragmented.

Configuring a GRE/IPv6 tunnel

Restrictions and guidelines

This task describes only GRE/IPv6 tunnel required tunnel interface commands (the **interface tunnel**, **source**, and **destination** commands). For more tunnel interface commands, see "Configuring tunneling."

Procedure

1. Enter system view.

   system-view
2. Create a GRE tunnel interface, and specify the tunnel mode as GRE/IPv6.

   interface tunnel *number* **mode gre ipv6**

   You must configure the same tunnel mode on both ends of a tunnel. Otherwise, packet delivery might fail.
3. Configure an IP address for the tunnel interface based on the passenger protocol.

   IPv4:

   For information about how to assign an IPv4 address to an interface, see "Configuring IP addressing."

   IPv6:

   For information about how to assign an IPv6 address to an interface, see "Configuring basic IPv6 settings."

   By default, no IP address is configured for a tunnel interface.
4. Configure a source IPv6 address or source interface for the tunnel.

   source { *ipv6-address* | *interface-type interface-number* }

   By default, no source IPv6 address or interface is configured for a tunnel.

   If you configure a source IPv6 address, the tunnel interface uses the source IPv6 address as the source IPv6 address of encapsulated packets.

   If you configure a source interface, the tunnel interface uses the IPv6 address of the source interface as the source IPv6 address of encapsulated packets.
5. Configure a destination IPv6 address for the tunnel.

   destination *ipv6-address*

   By default, no destination IPv6 address is configured for a tunnel.

The destination IPv6 address is the IPv6 address of the physical interface that the tunnel remote end uses to receive packets from the GRE tunnel.

The tunnel local end uses this address as the destination IPv6 address of encapsulated packets.

The tunnel destination address and the IP address of the tunnel interface must be in different subnets.

6. (Optional.) Configure GRE security mechanisms.
   - Enable GRE checksum.

     gre checksum

     By default, GRE checksum is disabled.
   - (Optional.) Configure a GRE key for the tunnel interface.

     gre key *key*

     By default, no GRE key is configured for a GRE tunnel interface.

     The two ends of a GRE tunnel must have the same key or both have no key.

Enabling dropping IPv6 packets that use IPv4-compatible IPv6 addresses

About this task

This feature enables the device to check the source and destination IPv6 addresses of the de-encapsulated IPv6 packets from a tunnel. If a packet uses an IPv4-compatible IPv6 address as the source or destination address, the device discards the packet.

Procedure

1. Enter system view.

   system-view
2. Configure the device to discard IPv6 packets with IPv4-compatible IPv6 addresses.

   tunnel discard ipv4-compatible-packet

   By default, the device does not discard such IPv6 packets.

   For more information about this command, see tunneling in *VPN Command Reference*.

Display and maintenance commands for GRE

Execute **display** commands in any view and **reset** commands in user view.

Task	Command	Remarks
Display information about tunnel interfaces.	**display interface** [**tunnel** [*number*]] [**brief** [**description** | **down**]]	For more information about the commands, see tunneling in *VPN Command Reference*.
Display IPv6 information about tunnel interface.	**display ipv6 interface** [**tunnel** [*number*]] [**brief**]	For more information about this command, see IPv6 basics in *Layer 3—IP Services Command Reference*.
Clear tunnel interface statistics.	**reset counters interface** [**tunnel** [*number*]]	For more information about this command, see tunneling in *VPN Command Reference*.

Task	Command	Remarks
Clear IPv6 statistics on tunnel interfaces.	**reset ipv6 statistics** [**slot** *slot-number*]	For more information about this command, see IPv6 basics in *Layer 3—IP Services Command Reference*.

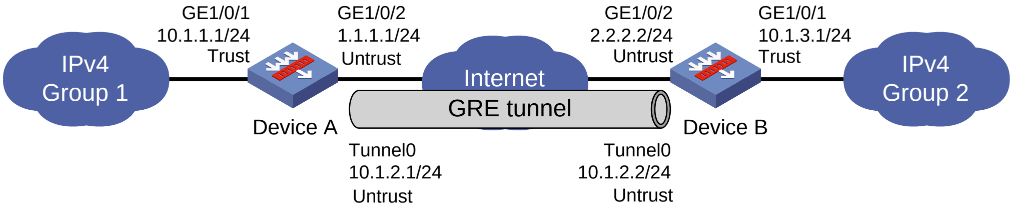
GRE configuration examples

Example: Configuring an IPv4 over IPv4 GRE tunnel

Network configuration

As shown in Figure 7, Group 1 and Group 2 are two private IPv4 networks. The two networks both use private network addresses and belong to the same VPN. Establish a GRE tunnel between Device A and Device B to interconnect the two private IPv4 networks Group 1 and Group 2.

Figure 7 Network diagram



Procedure

1. Configure Device A:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0**, and specify the tunnel mode as GRE/IPv4.

```
[DeviceA] interface tunnel 0 mode gre
[DeviceA-Tunnel0] ip address 10.1.2.1 255.255.255.0
[DeviceA-Tunnel0] source 1.1.1.1
[DeviceA-Tunnel0] destination 2.2.2.2
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 1.1.1.2.

```
[DeviceA] ip route-static 2.2.2.2 24 1.1.1.2
[DeviceA] ip route-static 10.1.3.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
```

```
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalout
[DeviceA-security-policy-ip-1-grelocalout] source-zone local
[DeviceA-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 10.1.2.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 10.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 2.2.2.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 10.1.2.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 10.1.3.1
[DeviceA-security-policy-ip-1-grelocalout] action pass
[DeviceA-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name grelocalin
[DeviceA-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ip-2-grelocalin] destination-zone local
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 2.2.2.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 10.1.2.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 10.1.3.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 1.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 10.1.2.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 10.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] action pass
[DeviceA-security-policy-ip-2-grelocalin] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.3.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE/IPv4.

```
[DeviceB] interface tunnel 0 mode gre
[DeviceB-Tunnel0] ip address 10.1.2.2 255.255.255.0
[DeviceB-Tunnel0] source 2.2.2.2
[DeviceB-Tunnel0] destination 1.1.1.1
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 2.2.2.3.

```
[DeviceB] ip route-static 1.1.1.1 24 2.2.2.3
[DeviceB] ip route-static 10.1.1.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule 1 name grelocalout
[DeviceB-security-policy-ip-1-grelocalout] source-zone local
[DeviceB-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 2.2.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 10.1.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 10.1.3.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 1.1.1.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 10.1.2.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-grelocalout] action pass
[DeviceB-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceB-security-policy-ip-2-grelocalin] destination-zone local
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 1.1.1.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 10.1.2.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 2.2.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 10.1.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 10.1.3.1
[DeviceB-security-policy-ip-2-grelocalin] action pass
[DeviceB-security-policy-ip-2-grelocalin] quit
[DeviceB-security-policy-ip] quit
```

Verifying the configuration

# Display tunnel interface information on Device A.

```
[DeviceA] display interface tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum transmission unit: 1476
Internet address: 10.1.2.1/24 (primary)
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel keepalive disabled
Tunnel TTL 255
Tunnel protocol/transport GRE/IP
    GRE key disabled
    Checksumming of GRE packets disabled
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Display tunnel interface information on Device B.

```
[DeviceB] display interface tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum transmission unit: 1476
Internet address: 10.1.2.2/24 (primary)
Tunnel source 2.2.2.2, destination 1.1.1.1
Tunnel keepalive disabled
Tunnel TTL 255
Tunnel protocol/transport GRE/IP
    GRE key disabled
    Checksumming of GRE packets disabled
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# From Device B, ping the IP address of GigabitEthernet 1/0/1 on Device A.

```
[DeviceB] ping -a 10.1.3.1 10.1.1.1
Ping 10.1.1.1 (10.1.1.1) from 10.1.3.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.1.1: icmp_seq=0 ttl=255 time=11.000 ms
56 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=4 ttl=255 time=0.000 ms

--- Ping statistics for 10.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/2.400/11.000/4.317 ms
```

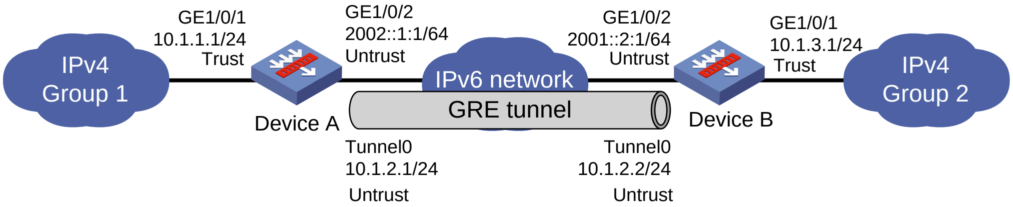
The output shows that Device B can successfully ping Device A.

Example: Configuring an IPv4 over IPv6 GRE tunnel

Network configuration

As shown in Figure 8, two IPv4 subnets Group 1 and Group 2 are connected to an IPv6 network. Create a GRE/IPv6 tunnel between Device A and Device B, so the two IPv4 subnets can communicate with each other through the GRE tunnel over the IPv6 network.

Figure 8 Network diagram



Procedure

1. Configure Device A:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipv6 address 2002::1.1/64
[DeviceA-GigabitEthernet1/0/2] quit
```

# Create tunnel interface **Tunnel 0**, and specify the tunnel mode as GRE/IPv6.

```
[DeviceA] interface tunnel 0 mode gre ipv6
[DeviceA-Tunnel0] ip address 10.1.2.1 255.255.255.0
[DeviceA-Tunnel0] source 2002::1:1
[DeviceA-Tunnel0] destination 2001::2:1
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 2002::1:2.

```
[DeviceA] ipv6 route-static 2001::2:1 64 2002::1:2
[DeviceA] ip route-static 10.1.3.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy and IPv6 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalout
[DeviceA-security-policy-ip-1-grelocalout] source-zone local
[DeviceA-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 10.1.2.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 10.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 10.1.2.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 10.1.3.1
[DeviceA-security-policy-ip-1-grelocalout] action pass
[DeviceA-security-policy-ip-1-grelocalout] quit
```

```
[DeviceA-security-policy-ip] quit
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name grelocalout
[DeviceA-security-policy-ipv6-1-grelocalout] source-zone local
[DeviceA-security-policy-ipv6-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ipv6-1-grelocalout] source-ip-host 2002::1:1
[DeviceA-security-policy-ipv6-1-grelocalout] destination-ip-host 2001::2:1
[DeviceA-security-policy-ipv6-1-grelocalout] action pass
[DeviceA-security-policy-ipv6-1-grelocalout] quit
[DeviceA-security-policy-ipv6] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy and IPv6 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalin
[DeviceA-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ip-2-grelocalin] destination-zone local
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 10.1.2.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 10.1.3.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 10.1.2.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 10.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] action pass
[DeviceA-security-policy-ip-2-grelocalin] quit
[DeviceA-security-policy-ip] quit
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name grelocalin
[DeviceA-security-policy-ipv6-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ipv6-2-grelocalin] destination-zone local
[DeviceA-security-policy-ipv6-2-grelocalin] source-ip-host 2001::2:1
[DeviceA-security-policy-ipv6-2-grelocalin] destination-ip-host 2002::1:1
[DeviceA-security-policy-ipv6-2-grelocalin] action pass
[DeviceA-security-policy-ipv6-2-grelocalin] quit
[DeviceA-security-policy-ipv6] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.3.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipv6 address 2001::2.1/64
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create tunnel interface **Tunnel 0**, and specify the tunnel mode as GRE/IPv6.

```
[DeviceB] interface tunnel 0 mode gre ipv6
[DeviceB-Tunnel0] ip address 10.1.2.2 255.255.255.0
[DeviceB-Tunnel0] source 2001::2:1
[DeviceB-Tunnel0] destination 2002::1:1
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 2001::2:2.

```
[DeviceB] ipv6 route-static 2002::1:1 64 2001::2:2
[DeviceB] ip route-static 10.1.1.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy and IPv6 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalout
[DeviceB-security-policy-ip-1-grelocalout] source-zone local
[DeviceB-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 10.1.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 10.1.3.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 10.1.2.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 10.1.1.1
[DeviceB-security-policy-ip-1-grelocalout] action pass
[DeviceB-security-policy-ip-1-grelocalout] quit
[DeviceB-security-policy-ip] quit
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name grelocalout
[DeviceB-security-policy-ipv6-1-grelocalout] source-zone local
[DeviceB-security-policy-ipv6-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-grelocalout] source-ip-host 2001::2:1
[DeviceB-security-policy-ipv6-1-grelocalout] destination-ip-host 2002::1:1
[DeviceB-security-policy-ipv6-1-grelocalout] action pass
[DeviceB-security-policy-ipv6-1-grelocalout] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy and IPv6 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceB-security-policy-ip-2-grelocalin] destination-zone local
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 10.1.2.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 10.1.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 10.1.3.1
[DeviceB-security-policy-ip-2-grelocalin] action pass
[DeviceB-security-policy-ip-2-grelocalin] quit
[DeviceB-security-policy-ip] quit
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name grelocalin
```

```
    [DeviceB-security-policy-ipv6-2-grelocalin] source-zone untrust
    [DeviceB-security-policy-ipv6-2-grelocalin] destination-zone local
    [DeviceB-security-policy-ipv6-2-grelocalin] source-ip-host 2002::1:1
    [DeviceB-security-policy-ipv6-2-grelocalin] destination-ip-host 2001::2:1
    [DeviceB-security-policy-ipv6-2-grelocalin] action pass
    [DeviceB-security-policy-ipv6-2-grelocalin] quit
    [DeviceB-security-policy-ipv6] quit
```

Verifying the configuration

# Display tunnel interface information on Device A.

```
[DeviceA] display interface tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum transmission unit: 1456
Internet address: 10.1.2.1/24 (primary)
Tunnel source 2002::1:1, destination 2001::2:1
Tunnel TTL 255
Tunnel protocol/transport GRE/IPv6
    GRE key disabled
    Checksumming of GRE packets disabled
Last clearing of counters: Never
Last 300 seconds input rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Last 300 seconds output rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Input: 10 packets, 840 bytes, 0 drops
Output: 10 packets, 840 bytes, 0 drops
```

# Display tunnel interface information on Device B.

```
[DeviceB] display interface tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum transmission unit: 1456
Internet address: 10.1.2.2/24 (primary)
Tunnel source 2001::2:1, destination 2002::1:1
Tunnel TTL 255
Tunnel protocol/transport GRE/IPv6
    GRE key disabled
    Checksumming of GRE packets disabled
Last clearing of counters: Never
Last 300 seconds input rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Last 300 seconds output rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Input: 10 packets, 840 bytes, 0 drops
Output: 10 packets, 840 bytes, 0 drops
```

# From Device B, ping the IP address of GigabitEthernet 1/0/1 on Device A.

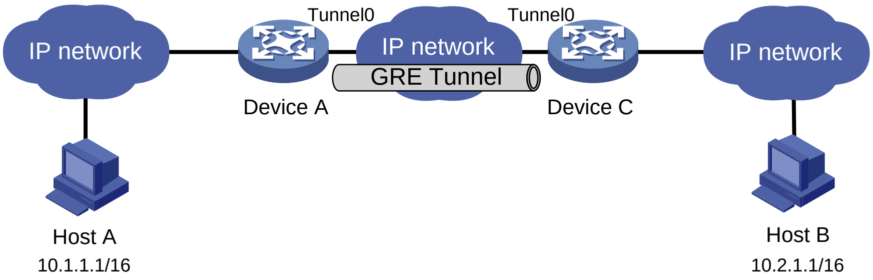
```
[DeviceB] ping -a 10.1.3.1 10.1.1.1
Ping 10.1.1.1 (10.1.1.1) from 10.1.3.1: 56 data bytes, press CTRL_C to break
56 bytes from 10.1.1.1: icmp_seq=0 ttl=255 time=2.000 ms
56 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=1.000 ms
56 bytes from 10.1.1.1: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 10.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.000/2.000/0.632 ms
```

The output shows that Device B can successfully ping Device A.

Troubleshooting GRE

The key to configuring GRE is to keep the configuration consistent. Most faults can be located by using the **`debugging gre`** or **`debugging tunnel`** command. This section analyzes one type of fault for illustration, with the scenario shown in Figure 9.

Figure 9 Network diagram



Hosts at both ends of a GRE tunnel cannot ping each other

Symptom

The interfaces at both ends of the tunnel are configured correctly and can ping each other, but Host A and Host B cannot ping each other.

Solution

To resolve the issue:

1. Execute the **`display ip routing-table`** command on Device A and Device C to view whether Device A has a route over tunnel 0 to 10.2.0.0/16 and whether Device C has a route over tunnel 0 to 10.1.0.0/16.
2. If such a route does not exist, execute the **`ip route-static`** command in system view to add the route. Take Device A as an example:

   ```
   [DeviceA] ip route-static 10.2.0.0 255.255.0.0 tunnel 0
   ```
3. If the issue persists, contact NSFOCUS Support.

Configuring GRE P2MP tunnels

About GRE P2MP tunnels

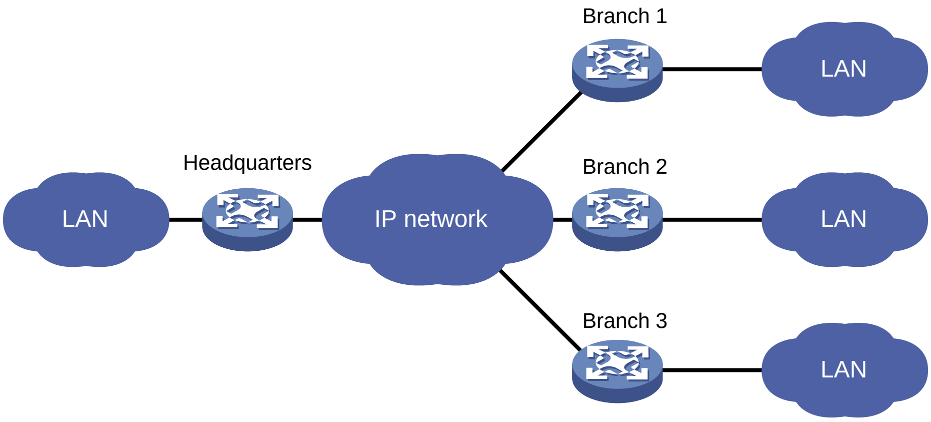
The GRE point-to-multipoint (P2MP) tunnel technology allows the device to use a GRE P2MP tunnel interface to establish a tunnel with multiple destinations.

Application scenario

As shown in Figure 10, an enterprise has multiple branches. For the headquarters to communicate with the branches through tunnels, you can deploy GRE P2MP.

Configure a GRE P2MP tunnel interface on the gateway of the headquarters. Configure GRE/IPv4 or GRE/IPv6 tunnel interfaces (also called point-to-point GRE tunnel interfaces) on the gateways of the branch networks. Use the GRE P2MP tunnel interface on the headquarters gateway to set up a point-to-multipoint tunnel with the point-to-point GRE tunnel interfaces on the branch gateways.

Figure 10 GRE P2MP tunnel application



Benefits

Compared with GRE point-to-point tunnels and other dynamic VPN technologies, GRE P2MP has the following benefits:

- **Simplified configuration**—One GRE P2MP tunnel interface can set up a point-to-multipoint tunnel with multiple point-to-point GRE tunnel interfaces. You do not need to create multiple GRE tunnel interfaces for the headquarters gateway to establish point-to-point tunnels with each branch.
- **Easy maintenance**—When a new branch is added, the headquarters gateway can dynamically learn the tunnel destination address through GRE packets received from the branch. The gateway can automatically set up a tunnel with the branch. Manual configuration is not required.
- **Flexible access**—The headquarters can dynamically learn the tunnel destination addresses. Whether the branch network obtains public addresses dynamically or not does not affect the configuration in the headquarters.
- **Good interoperability and decreased TCO**—The device implements GRE P2MP based on the standard GRE protocol. It does not require other proprietary protocols. GRE P2MP does not have special requirements for the branch gateways. You can use any devices that support GRE as branch gateways.

- **High availability**—Support for tunnel backup in the headquarters and branches.

Working mechanisms

GRE uses the same packet encapsulation and de-encapsulation processes for point-to-multipoint and point-to-point tunnels.

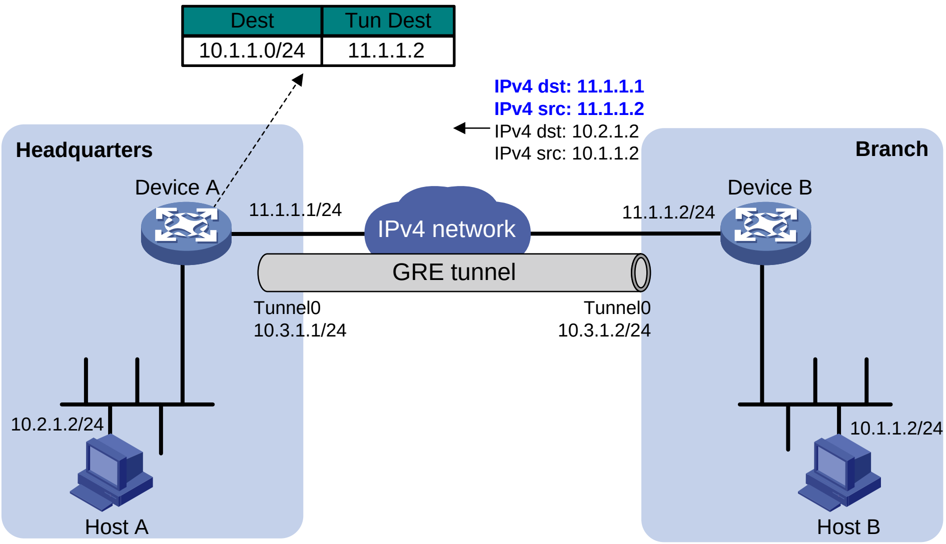
When the device uses a GRE P2MP tunnel to forward traffic, it searches the tunnel entries for the tunnel destination address based on the packet destination IP address. GRE encapsulates the tunnel destination address as the destination IP address in the GRE transport protocol header.

GRE P2MP tunnel entries include dynamic entries and static entries. Dynamic and static entries cannot coexist.

GRE P2MP dynamic tunnel entry learning

As shown in Figure 11, Device A dynamically learns GRE P2MP tunnel entry information (tunnel destination address and packet destination address) from the GRE packets received from the branch. The tunnel destination address is the source IP address in the transport protocol header of the received GRE packet. The packet destination address (branch network address) is the source IP address in passenger protocol header of the received GRE packet.

Figure 11 Dynamically learning a GRE P2MP tunnel entry



GRE P2MP static tunnel entry configuration

In a dynamic GRE P2MP tunnel network, the headquarters gateway cannot actively send packets to branch networks. If the gateway does not receive GRE packets from a branch, it cannot create a dynamic tunnel entry for the branch. To resolve this issue, you can configure GRE P2MP static tunnel entries by applying a GRE P2MP tunnel template to the GRE P2MP tunnel interface. The tunnel mapping entries in the tunnel template are static tunnel entries. A static tunnel entry includes the following information:

- Tunnel destination address.
- Branch network address.
- VPN instance of the branch network.

In the current software version, the device supports only IPv4 static tunnel entries for GRE P2MP.

GRE P2MP tunnel backup

This feature applies only to dynamic GRE P2MP tunnels.

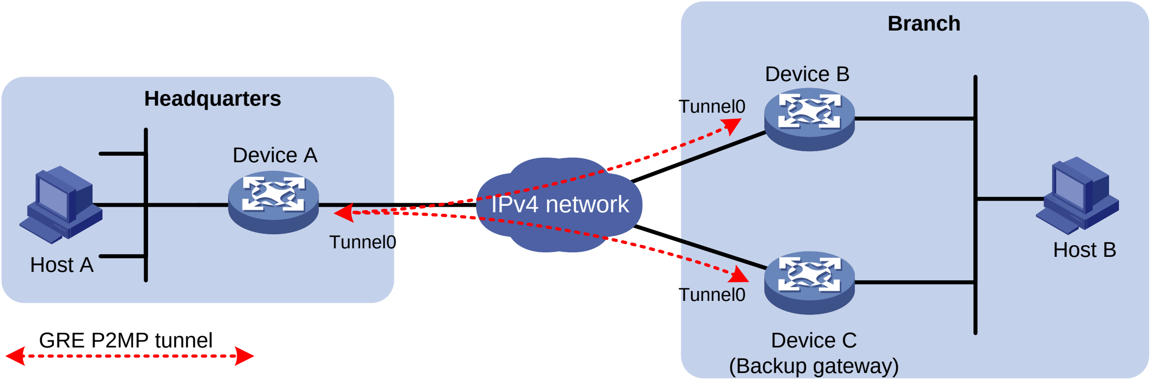
GRE tunnel backup in branches

As shown in Figure 12, deploy redundant gateways in the branch network. Configure a GRE P2MP tunnel interface on the gateway of the headquarters. Use the GRE P2MP tunnel interface to set up a dynamic P2MP tunnel with each branch gateway.

Assign different GRE keys to the tunnel interfaces on the branch gateways. The headquarters gateway selects a dynamic tunnel entry for the branch in the following order:

1. The dynamic tunnel entry that does not have a GRE key.
2. The dynamic tunnel entry that has a GRE key with a lower value.
3. The latest learned dynamic tunnel entry.

Figure 12 GRE tunnel backup in a branch



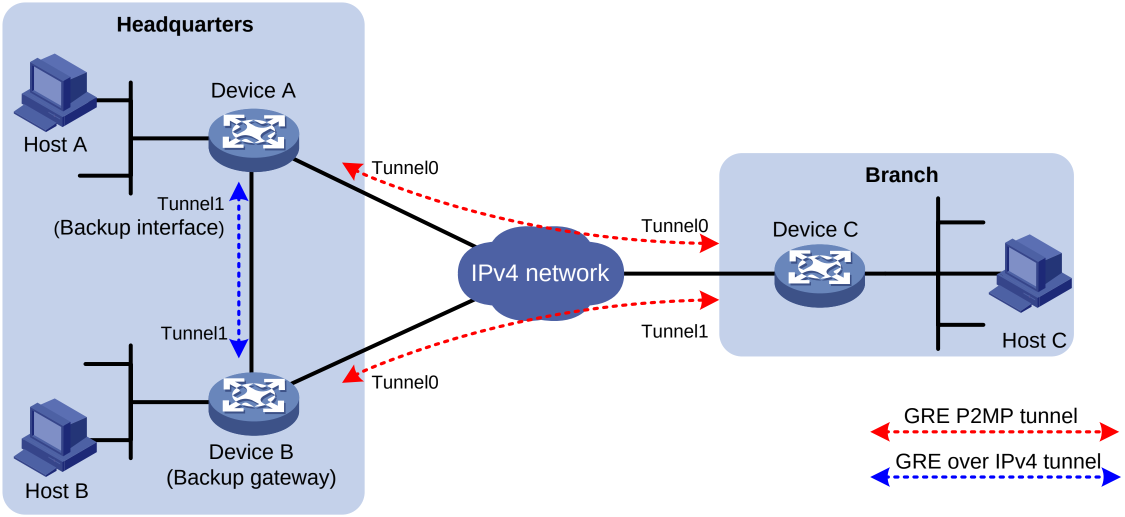
GRE tunnel backup in the headquarters

As shown in Figure 13, deploy redundant gateways in the headquarters, one primary and one backup. Configure a GRE P2MP tunnel interface and a GRE/IPv4 or GRE/IPv6 tunnel interface on each gateway. Specify the GRE/IPv4 or GRE/IPv6 tunnel interface on the primary gateway as the backup interface of its GRE P2MP tunnel interface.

When the primary gateway cannot reach the branch network, it sends the traffic destined for the branch network to the backup gateway through the backup interface. Then, the backup gateway sends the traffic to the destination.

The backup interface is also included for tunnel selection on the primary gateway. If the backup interface does not have a GRE key, its priority is lower than all GRE P2MP dynamic tunnel entries. If the backup interface has a GRE key, the headquarters gateway compares the GRE key with the GRE keys of GRE P2MP dynamic tunnel entries. The lower the value, the higher the priority.

Figure 13 Tunnel backup in the headquarters



Restrictions: Hardware compatibility with GRE P2MP tunnels

Models	GRE P2MP tunnel compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080	Yes
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: GRE P2MP tunnel configuration

For more information about the **`interface tunnel`**, **`source`**, and **`tunnel discard ipv4-compatible-packet`** commands and additional configuration commands on a tunnel interface, see "Configuring tunneling" and tunneling commands in *VPN Command Reference*.

Restrictions and guidelines for the headquarters end of a P2MP tunnel

When you configure the headquarters end of a P2MP tunnel, follow these restrictions and guidelines:

- Do not configure the same tunnel source address for multiple P2MP tunnel interfaces on the gateway.
- Do not configure the same tunnel source and destination addresses for a GRE P2MP tunnel interface and a GRE tunnel interface.
- Do not configure a GRE key on a GRE P2MP tunnel interface.
- You can specify a mask or prefix length for the headquarters gateway to create dynamic tunnel entries for branch networks. The gateway generates only one dynamic tunnel entry for each branch. Make sure the gateway can reach the devices in all branches by using the dynamic tunnel entries.

Restrictions and guidelines for the branch ends of a P2MP tunnel

The branch networks cannot communicate with one another. They do not have tunnels to reach one another.

When you configure the branch ends of a P2MP tunnel, follow these restrictions and guidelines:

- The tunnel destination address at each branch end must be the tunnel source address configured on the P2MP tunnel interface on the headquarters.
- On each gateway, the IP address of the tunnel interface and the tunnel destination address configured on the tunnel interface must be in different subnets.
- You can configure GRE keys on the branch GRE tunnel interfaces for the headquarters to determine the dynamic tunnel entry priority of each branch.

Restrictions and guidelines for all ends of a P2MP tunnel

The following restrictions and guidelines apply to all ends of a P2MP tunnel:

- You can enable or disable GRE checksum at each end of a tunnel. If GRE checksum is enabled at a tunnel end, the tunnel end sends packets carrying the checksum to the peer end. A tunnel end checks the GRE checksum of a received packet if the packet carries a GRE checksum, whether or not the tunnel end is enabled with GRE checksum.
- Configure routes reaching the destination networks through the local tunnel interface. You can configure the routes by using one of the following methods:
  - Configure static routes, using the local tunnel interface as the outgoing interface of the routes.
  - Enable a dynamic routing protocol on both the tunnel interface and the interface connecting the private network. This allows the dynamic routing protocol to establish routing entries with the tunnel interface as the outgoing interface.
- Dynamic entry-based GRE P2MP tunnels cannot forward private traffic.

GRE P2MP tunnel interface configuration tasks at a glance

To configure a GRE P2MP tunnel interface, perform one of the following tasks:

- Configuring a GRE P2MP tunnel interface for dynamic entry-based tunnel setup
- Configuring a GRE P2MP tunnel interface for static entry-based tunnel setup

Configuring a GRE P2MP tunnel interface

Configuring a GRE P2MP tunnel interface for dynamic entry-based tunnel setup

1. Enter system view.

   `system-view`

2. Create a GRE P2MP tunnel interface and enter tunnel interface view.

   interface tunnel *interface-number* **mode gre-p2mp** [**ipv6**]

   In this mode, the transport and passenger protocols support both IPv4 and IPv6.

   The other end of a GRE P2MP tunnel interface must be a GRE/IPv4 or GRE/IPv6 tunnel interface.

3. Configure an IP address for the tunnel interface.

IPv4:

For information about how to assign an IPv4 address to an interface, see "Configuring IP addressing."

IPv6:

For information about how to assign an IPv6 address to an interface, see "Configuring basic IPv6 settings."

By default, no IP address is configured for a tunnel interface.

4. Configure a source IP address or source interface for the tunnel.

   ```
   source { ipv4-address | ipv6-address | interface-type
   interface-number }
   ```

   By default, no source IP address or interface is configured for a tunnel.

   If you configure a source IP address, the tunnel interface uses the source IP address as the source IP address of encapsulated packets.

   If you configure a source interface, the tunnel interface uses the following IP address as the source IP address of encapsulated packets:

   - Primary IP address of the source interface in an IPv4 network.
   - The lowest IP address of the source interface in an IPv6 network.

5. (Optional.) Enable GRE checksum.

   ```
   gre checksum
   ```

   By default, GRE checksum is disabled.

6. (Optional.) Set an aging timer for dynamic tunnel entries of the GRE P2MP tunnel interface.

   ```
   gre p2mp aging-time aging-time
   ```

   By default, the aging timer is 5 seconds for dynamic tunnel entries of a GRE P2MP tunnel interface.

7. (Optional.) Specify a backup interface for the GRE P2MP tunnel interface.

   ```
   gre p2mp backup-interface tunnel number
   ```

   By default, no backup interface is specified for a GRE P2MP tunnel interface.

   The backup interface must be a GRE/IPv4 or GRE/IPv6 tunnel interface. The backup interface must exist.

8. (Optional.) Specify an IPv4 address mask or IPv6 address prefix for the branch networks.

   ```
   gre p2mp branch-network-mask { mask | mask-length | ipv6 prefix-length }
   ```

   By default, the IPv4 address mask is 255.255.255.255 (the mask length is 32) and the IPv6 address prefix length is 128.

9. (Optional.) Enable next hop host route learning.

   ```
   gre p2mp nexthop-learning
   ```

   By default, next hop host route learning is disabled.

   This command is supported only on GRE/IPv4 P2MP tunnel interfaces.

10. Return to system view.

    ```
    quit
    ```

11. (Optional.) Enable dropping IPv6 packets that use IPv4-compatible IPv6 addresses.

    ```
    tunnel discard ipv4-compatible-packet
    ```

    By default, IPv6 packets that use IPv4-compatible IPv6 packets are not dropped.

Configuring a GRE P2MP tunnel interface for static entry-based tunnel setup

1. Enter system view.

   system-view

2. Create a GRE P2MP tunnel template and enter its view.

   gre p2mp-template *template-name*

3. Configure a tunnel mapping entry.

   map [**vpn-instance** *vpn-instance-name*] **branch-network-address** *branch-network-address* { *mask* | *mask-length* } **tunnel-destination** *tunnel-dest-address* [**checksum-fill** *checksum-value*]

   By default, no tunnel mapping entries are configured for a GRE P2MP tunnel template.

   Specify the **checksum-fill** *checksum-value* option depending on your network. The option might cause GRE checksum failures.

4. Return to system view.

   quit

5. Create a GRE P2MP tunnel interface and enter its view.

   interface tunnel *interface-number* **mode gre-p2mp**

   The other end of a GRE P2MP tunnel interface must be a GRE/IPv4 tunnel interface.

6. Configure an IPv4 address for the tunnel interface.

   For information about how to assign an IPv4 address to an interface, see "Configuring IP addressing."

   By default, no IPv4 address is configured for a tunnel interface.

7. Configure a source IP address or source interface for the tunnel.

   source { *ipv4-address* | *interface-type interface-number* }

   By default, no source IP address or interface is configured for a tunnel.

   If you configure a source IP address, the tunnel interface uses the source IP address as the source IP address of encapsulated packets.

   If you configure a source interface, the tunnel interface uses the primary IP address of the source interface as the source IP address of encapsulated packets.

8. Apply the GRE P2MP tunnel template to the tunnel interface.

   gre p2mp-template *template-name*

   By default, no GRE P2MP tunnel template is applied to a GRE P2MP tunnel interface.

   You must specify an existing GRE P2MP tunnel template for this command.

9. (Optional.) Enable GRE checksum.

   gre checksum

   By default, GRE checksum is disabled.

10. (Optional.) Enable GRE packets to carry VPC information.

    gre vpc enable

    By default, GRE packets do not carry VPC information.

11. (Optional.) Configure the preference of static routes issued by the tunnel mapping entries in the GRE P2MP tunnel template.

    tunnel route-static [**preference** *preference-value*]

    By default, the preference is 60.

Display and maintenance commands for GRE P2MP tunnels

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display dynamic tunnel entry information for a GRE P2MP tunnel interface.	**display gre p2mp tunnel-table interface tunnel** *number* [**ipv4** | **ipv6**]
Display packet statistics of static tunnel entries for a GRE P2MP tunnel interface.	**display gre p2mp tunnel-table statistics interface tunnel** *number* [**vpn-instance** *vpn-instance-name*] [**branch-network-address** *branch-network-address* { *mask* | *mask-length* }]
Clear dynamic tunnel entry information for a GRE P2MP tunnel interface.	**reset gre p2mp tunnel-table interface tunnel** *number* [**destination** { *dest-address* | **ipv6** *dest-ipv6-address* } **tunnel-destination** { *tunnel-dest-address* | **ipv6** *tunnel-dest-address* }]
Clear packet statistics of static tunnel entries for a GRE P2MP tunnel interface.	**reset gre p2mp tunnel-table statistics interface tunnel** *number* [**vpn-instance** *vpn-instance-name*] [**branch-network-address** *branch-network-address* { *mask* | *mask-length* }]

GRE P2MP tunnel configuration examples

Example: Configuring a GRE P2MP tunnel

Network configuration

As shown in Figure 14, Device A is the gateway of the headquarters, and Device B and Device C are the gateways of the branches.

Configure the devices to meet the following requirements:

- Device A can dynamically set up a GRE P2MP tunnel with multiple branch gateways.
- Device B and Device C each can set up a GRE tunnel with Device A.
- Branch 1 and Branch 2 cannot communicate.

Figure 14 Network diagram

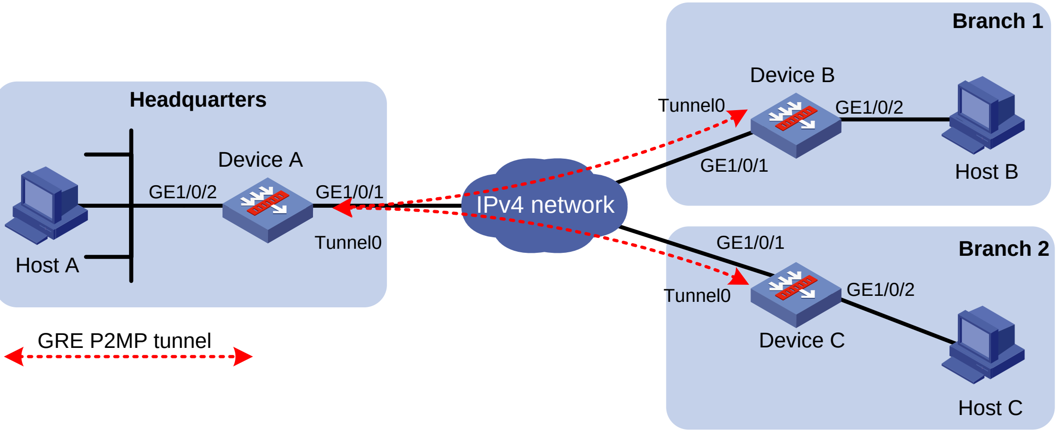


Table 1 IP address assignment

Device	Interface	IP address	Security zone
Device A	GE1/0/1	11.1.1.1/24	Untrust
	GE1/0/2	192.168.11.1/24	Trust
	Tunnel 0	192.168.22.1/24	Untrust
Device B	GE1/0/1	11.1.2.2/24	Untrust
	GE1/0/2	192.168.12.1/24	Trust
	Tunnel 0	192.168.22.2/24	Untrust
Device C	GE1/0/1	11.1.2.3/24	Untrust
	GE1/0/2	192.168.13.1/24	Trust
	Tunnel 0	192.168.22.3/24	Untrust

Procedure

This example uses Device B to describe the gateway configuration for a branch. The gateway configuration for other branches is the same as the gateway configuration on Device B.

1. Configure Device A:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 11.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE P2MP.

```
[DeviceA] interface tunnel 0 mode gre-p2mp
[DeviceA-Tunnel0] ip address 192.168.22.1 255.255.255.0
[DeviceA-Tunnel0] source 11.1.1.1
[DeviceA-Tunnel0] gre p2mp branch-network-mask 255.255.255.0
[DeviceA-Tunnel0] gre p2mp aging-time 20
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.2.2.

```
[DeviceA] ip route-static 11.1.2.2 24 11.1.1.2
[DeviceA] ip route-static 192.168.12.0 255.255.255.0 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalout
[DeviceA-security-policy-ip-1-grelocalout] source-zone local
[DeviceA-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 11.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 192.168.22.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 192.168.11.1
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 11.1.2.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.12.1
[DeviceA-security-policy-ip-1-grelocalout] action pass
[DeviceA-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ip] rule name grelocalin
[DeviceA-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ip-2-grelocalin] destination-zone local
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 11.1.2.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.12.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 11.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 192.168.22.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 192.168.11.1
[DeviceA-security-policy-ip-2-grelocalin] action pass
[DeviceA-security-policy-ip-2-grelocalin] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 11.1.2.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE/IPv4.

```
[DeviceB] interface tunnel 0 mode gre
[DeviceB-Tunnel0] ip address 192.168.22.2 255.255.255.0
```

```
[DeviceB-Tunnel0] source 11.1.2.2
[DeviceB-Tunnel0] destination 11.1.1.1
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.2.3.

```
[DeviceB] ip route-static 11.1.1.1 24 11.1.2.3
[DeviceB] ip route-static 192.168.11.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalout
[DeviceB-security-policy-ip-1-grelocalout] source-zone local
[DeviceB-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 11.1.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 192.168.22.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 192.168.12.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 11.1.1.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 192.168.11.1
[DeviceB-security-policy-ip-1-grelocalout] action pass
[DeviceB-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceB-security-policy-ip-2-grelocalin] destination-zone local
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 11.1.1.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 192.168.11.1
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 11.1.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 192.168.22.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 192.168.12.1
[DeviceB-security-policy-ip-2-grelocalin] action pass
[DeviceB-security-policy-ip-2-grelocalin] quit
[DeviceB-security-policy-ip] quit
```

Verifying the configuration

# Display GRE P2MP tunnel entries on Device A. The output shows that Device A does not have GRE P2MP tunnel entries.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:0
```

```
Dest Addr               Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
```

# Ping Host A from Host B. The ping operation succeeds. (Details not shown.)

# Display GRE P2MP tunnel entries again on Device A. The output shows that Device A has created a GRE P2MP tunnel entry destined for the branch.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:1
Dest Addr               Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.12.0            255.255.255.0   11.1.2.2                          5
```

Example: Configuring GRE P2MP headquarters-side tunnel backup

Network configuration

As shown in Figure 15, Device A is the gateway of the headquarters, Device B is the backup gateway of the headquarters, and Device C is the gateway of the branch.

Configure the devices as follows:

- Configure a GRE P2MP tunnel interface on Device A and Device B, respectively. Device A and Device B use the interfaces to establish GRE P2MP tunnels with Device C.
- Configure a GRE/IPv4 tunnel interface on Device A and Device B, respectively. Configure the GRE/IPv4 tunnel interface on Device A as the backup interface of the GRE P2MP tunnel interface. To avoid loops, do not configure the GRE/IPv4 tunnel interface on Device B as the backup interface of the GRE P2MP tunnel interface.
- Configure GRE/IPv4 tunnel interfaces on Device C for the device to establish tunnels with Device A and Device B.

Figure 15 Network diagram

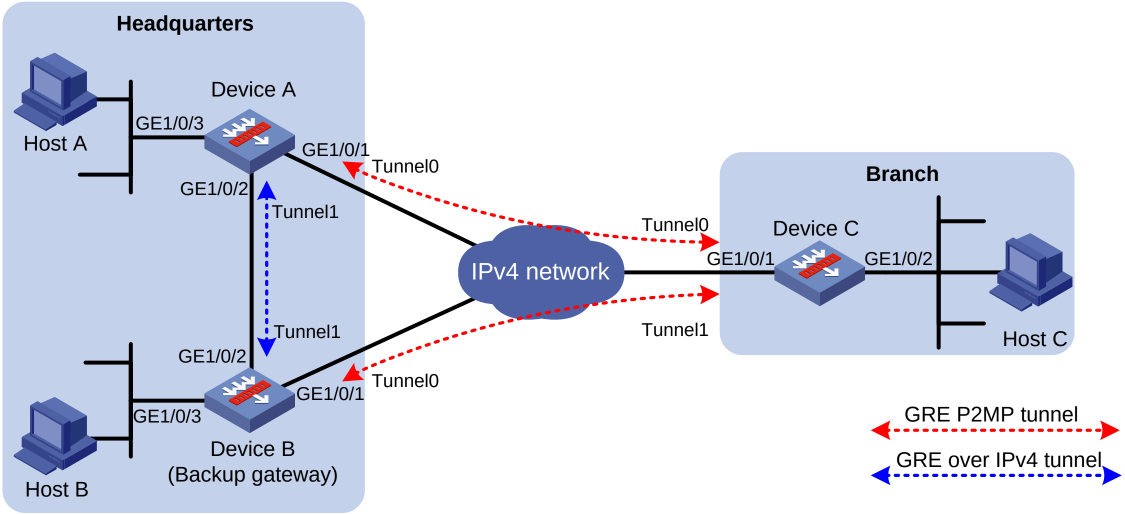


Table 2 IP address assignment

Device	Interface	IP address	Security zone
Device A	GE1/0/1	11.1.1.1/24	Untrust
	GE1/0/2	10.1.1.1/24	Trust
	GE1/0/3	192.168.11.1/24	Trust

Device	Interface	IP address	Security zone
	Tunnel 0	172.168.1.1/24	Untrust
	Tunnel 1	192.168.22.1/24	Trust
Device B	GE1/0/1	11.1.2.2/24	Untrust
	GE1/0/2	10.1.1.2/24	Trust
	GE1/0/3	192.168.11.2/24	Trust
	Tunnel 0	172.168.2.2/24	Untrust
	Tunnel 1	192.168.22.2/24	Trust
Device C	GE1/0/1	11.1.2.3/24	Untrust
	GE1/0/2	192.168.12.1/24	Trust
	Tunnel 0	172.168.1.3/24	Untrust
	Tunnel 1	172.168.2.3/24	Untrust

Restrictions and guidelines

You need to disconnect the link between Device A and Device C to verify the tunnel backup feature. To make sure Device C can detect the link loss and switch the traffic forwarding path to the link connecting Device B, perform one of the following tasks:

- If Device C and Device A are directly connected, configure static routes on Device C.
- If Device C and Device A are not directly connected, use one of the following methods:
  - Configure a dynamic routing protocol on Device A, Device B, and Device C.
  - Configure static routes on Device C and associate the routes with track entries for detecting the route reachability status. For more information about track entries, see Track in *Network Management and Monitoring Configuration Guide*.

This example uses the static routes method.

Procedure

1. Configure Device A:

   # Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 11.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   # Create tunnel interface **Tunnel 1** and specify the tunnel mode as GRE/IPv4.

```
[DeviceA] interface tunnel 1 mode gre
[DeviceA-Tunnel1] ip address 192.168.22.1 255.255.255.0
[DeviceA-Tunnel1] source 10.1.1.1
[DeviceA-Tunnel1] destination 10.1.2.2
[DeviceA-Tunnel1] quit
```

   # Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE P2MP.

```
[DeviceA] interface tunnel 0 mode gre-p2mp
[DeviceA-Tunnel0] ip address 172.168.1.1 255.255.255.0
[DeviceA-Tunnel0] source 11.1.1.1
[DeviceA-Tunnel0] gre p2mp branch-network-mask 255.255.255.0
```

```
[DeviceA-Tunnel0] gre p2mp aging-time 20
[DeviceA-Tunnel0] gre p2mp backup-interface tunnel 1
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.1.2.

```
[DeviceA] ip route-static 11.1.3.3 24 11.1.1.2
[DeviceA] ip route-static 192.168.12.0 255.255.255.0 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] import interface Tunnel 1
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device A to send packets to Device C.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalout
[DeviceA-security-policy-ip-1-grelocalout] source-zone local
[DeviceA-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 11.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 172.168.1.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 192.168.11.1
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 11.1.3.3
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 172.168.1.3
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.12.1
[DeviceA-security-policy-ip-1-grelocalout] action pass
[DeviceA-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device C.

```
[DeviceA-security-policy-ip] rule name grelocalin
[DeviceA-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ip-2-grelocalin] destination-zone local
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 11.1.3.3
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 172.168.1.3
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.12.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 11.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 172.168.1.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 192.168.11.1
[DeviceA-security-policy-ip-2-grelocalin] action pass
[DeviceA-security-policy-ip-2-grelocalin] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device A to send packets to Device B.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name grelocalout
```

```
[DeviceA-security-policy-ip-3-grelocalout] source-zone local
[DeviceA-security-policy-ip-3-grelocalout] destination-zone trust
[DeviceA-security-policy-ip-3-grelocalout] source-ip-host 10.1.1.1
[DeviceA-security-policy-ip-3-grelocalout] source-ip-host 192.168.22.1
[DeviceA-security-policy-ip-3-grelocalout] source-ip-host 192.168.11.1
[DeviceA-security-policy-ip-3-grelocalout] destination-ip-host 10.1.1.2
[DeviceA-security-policy-ip-3-grelocalout] destination-ip-host 192.168.22.2
[DeviceA-security-policy-ip-3-grelocalout] destination-ip-host 192.168.12.1
[DeviceA-security-policy-ip-3-grelocalout] action pass
[DeviceA-security-policy-ip-3-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B.

```
[DeviceA-security-policy-ip] rule 4 name grelocalin
[DeviceA-security-policy-ip-4-grelocalin] source-zone trust
[DeviceA-security-policy-ip-4-grelocalin] destination-zone local
[DeviceA-security-policy-ip-4-grelocalin] source-ip-host 10.1.1.2
[DeviceA-security-policy-ip-4-grelocalin] source-ip-host 192.168.22.2
[DeviceA-security-policy-ip-4-grelocalin] source-ip-host 192.168.12.1
[DeviceA-security-policy-ip-4-grelocalin] destination-ip-host 10.1.1.1
[DeviceA-security-policy-ip-4-grelocalin] destination-ip-host 192.168.22.1
[DeviceA-security-policy-ip-4-grelocalin] destination-ip-host 192.168.11.1
[DeviceA-security-policy-ip-4-grelocalin] action pass
[DeviceA-security-policy-ip-4-grelocalin] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 11.1.2.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE P2MP.

```
[DeviceB] interface tunnel 0 mode gre-p2mp
[DeviceB-Tunnel0] ip address 172.168.2.2 255.255.255.0
[DeviceB-Tunnel0] source 11.1.2.2
[DeviceB-Tunnel0] gre p2mp branch-network-mask 255.255.255.0
[DeviceB-Tunnel0] gre p2mp aging-time 20
[DeviceB-Tunnel0] quit
```

# Create tunnel interface **Tunnel 1** and specify the tunnel mode as GRE/IPv4.

```
[DeviceB] interface tunnel 1 mode gre
[DeviceB-Tunnel1] ip address 192.168.22.2 255.255.255.0
[DeviceB-Tunnel1] source 10.1.1.2
[DeviceB-Tunnel1] destination 10.1.1.1
[DeviceB-Tunnel1] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.2.3.

```
[DeviceB] ip route-static 11.1.3.3 24 11.1.2.3
[DeviceB] ip route-static 192.168.12.1 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-trust] import interface Tunnel 1
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device B to send packets to Device C.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalout
[DeviceB-security-policy-ip-1-grelocalout] source-zone local
[DeviceB-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 11.1.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 172.168.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 192.168.11.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 11.1.3.3
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 172.168.2.3
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 192.168.12.1
[DeviceB-security-policy-ip-1-grelocalout] action pass
[DeviceB-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device C.

```
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceB-security-policy-ip-2-grelocalin] destination-zone local
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 11.1.3.3
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 172.168.2.3
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 192.168.12.1
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 11.1.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 172.168.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 192.168.11.1
[DeviceB-security-policy-ip-2-grelocalin] action pass
[DeviceB-security-policy-ip-2-grelocalin] quit
[DeviceB-security-policy-ip] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalout
[DeviceB-security-policy-ip-3-grelocalout] source-zone local
[DeviceB-security-policy-ip-3-grelocalout] destination-zone trust
[DeviceB-security-policy-ip-3-grelocalout] source-ip-host 10.1.1.2
[DeviceB-security-policy-ip-3-grelocalout] source-ip-host 192.168.22.2
[DeviceB-security-policy-ip-3-grelocalout] source-ip-host 192.168.12.1
[DeviceB-security-policy-ip-3-grelocalout] destination-ip-host 10.1.1.1
[DeviceB-security-policy-ip-3-grelocalout] destination-ip-host 192.168.22.1
```

```
[DeviceB-security-policy-ip-3-grelocalout] destination-ip-host 192.168.11.1
[DeviceB-security-policy-ip-3-grelocalout] action pass
[DeviceB-security-policy-ip-3-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-4-grelocalin] source-zone trust
[DeviceB-security-policy-ip-4-grelocalin] destination-zone local
[DeviceB-security-policy-ip-4-grelocalin] source-ip-host 10.1.1.1
[DeviceB-security-policy-ip-4-grelocalin] source-ip-host 192.168.22.1
[DeviceB-security-policy-ip-4-grelocalin] source-ip-host 192.168.11.1
[DeviceB-security-policy-ip-4-grelocalin] destination-ip-host 10.1.1.2
[DeviceB-security-policy-ip-4-grelocalin] destination-ip-host 192.168.22.2
[DeviceB-security-policy-ip-4-grelocalin] destination-ip-host 192.168.12.1
[DeviceB-security-policy-ip-4-grelocalin] action pass
[DeviceB-security-policy-ip-4-grelocalin] quit
[DeviceB-security-policy-ip] quit
```

3. Configure Device C:

# Assign an IP addresse to interface GigabitEthernet 1/0/1.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ip address 11.1.3.3 255.255.255.0
[DeviceC-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE/IPv4.

```
[DeviceC] interface tunnel 0 mode gre
[DeviceC-Tunnel0] ip address 172.168.1.3 255.255.255.0
[DeviceC-Tunnel0] source 11.1.3.3
[DeviceC-Tunnel0] destination 11.1.1.1
[DeviceC-Tunnel0] quit
```

# Create tunnel interface **Tunnel 1** and specify the tunnel mode as GRE/IPv4.

```
[DeviceC] interface tunnel 1 mode gre
[DeviceC-Tunnel1] ip address 172.168.2.3 255.255.255.0
[DeviceC-Tunnel1] source 11.1.3.3
[DeviceC-Tunnel1] destination 11.1.2.2
[DeviceC-Tunnel1] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 or Tunnel 1 and the next hop is 11.1.3.4. The route with priority 1 has a higher priority than the route with priority 10.

```
[DeviceC] ip route-static 11.1.1.1 24 11.1.3.4
[DeviceC] ip route-static 11.1.2.2 24 11.1.3.4
[DeviceC] ip route-static 192.168.11.0 255.255.255.0 tunnel 0 preference 1
[DeviceC] ip route-static 192.168.11.0 255.255.255.0 tunnel 1 preference 10
```

# Add interfaces to security zones.

```
[DeviceC] security-zone name Untrust
[DeviceC-security-zone-Untrust] import interface Tunnel 0
[DeviceC-security-zone-Untrust] import interface Tunnel 1
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
```

```
[DeviceC-security-zone-Untrust] quit
[DeviceC] security-zone name Trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device C to send packets to Device A and Device B.

```
[DeviceC] security-policy ip
[DeviceC-security-policy-ip] rule name grelocalout
[DeviceC-security-policy-ip-1-grelocalout] source-zone local
[DeviceC-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 11.1.3.3
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 172.168.1.3
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 172.168.2.3
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 192.168.12.1
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 11.1.1.1
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 11.1.2.2
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 172.168.1.1
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.2
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 192.168.11.1
[DeviceC-security-policy-ip-1-grelocalout] action pass
[DeviceC-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device C to receive the packets sent from Device A and Device B.

```
[DeviceC-security-policy-ip] rule name grelocalin
[DeviceC-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceC-security-policy-ip-2-grelocalin] destination-zone local
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 11.1.1.1
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 11.1.2.2
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 172.168.1.1
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.2
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 192.168.11.1
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 11.1.3.3
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 172.168.1.3
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 172.168.2.3
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 192.168.12.1
[DeviceC-security-policy-ip-2-grelocalin] action pass
[DeviceC-security-policy-ip-2-grelocalin] quit
[DeviceC-security-policy-ip] quit
```

Verifying the configuration

# Ping Host A from Host C. The ping operation succeeds. (Details not shown.)

# Display GRE P2MP tunnel entries on Device A.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:1
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.12.0          255.255.255.0   11.1.3.3                          10
```

The output shows that Device A has a GRE P2MP tunnel entry for the branch.

# Shut down tunnel interface **Tunnel 0** on Device C.

```
[DeviceC] interface tunnel 0
[DeviceC-Tunnel0] shutdown
[DeviceC-Tunnel0] quit
```

# Verify that Device A has deleted the GRE P2MP tunnel entry for the branch after the tunnel entry aging timer expires.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:0
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
```

# Ping Host A from Host C. The ping operation succeeds. (Details not shown.)

# Display GRE P2MP tunnel entries on Device B.

```
[DeviceB] display gre p2mp tunnel-table interface tunnel 0
Total number:1
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.12.0          255.255.255.0   11.1.3.3                          10
```

The output shows that Device B has created a tunnel entry destined for the branch. This indicates that Device A forwarded traffic destined for the branch to Device B through the backup tunnel interface.

Example: Configuring GRE P2MP branch-side tunnel backup

Network configuration

As shown in Figure 16, Device A is the gateway of the headquarters, Device B is the gateway of the branch, and Device C is the backup gateway of the branch.

Configure the devices as follows:

- Configure a GRE P2MP tunnel interface on Device A. Device A uses the interface to establish a GRE P2MP tunnel with Device B and Device C.
- Configure a GRE/IPv4 tunnel interface on Device B and Device C, respectively. The devices use the tunnel interfaces to establish tunnels with Device A.
- Assign different GRE keys to the GRE/IPv4 tunnel interfaces on Device B and Device C. Device A selects the tunnel that has a GRE key with a higher priority to forward traffic. In this example, the tunnel destined for Device B has a higher priority.

Figure 16 Network diagram

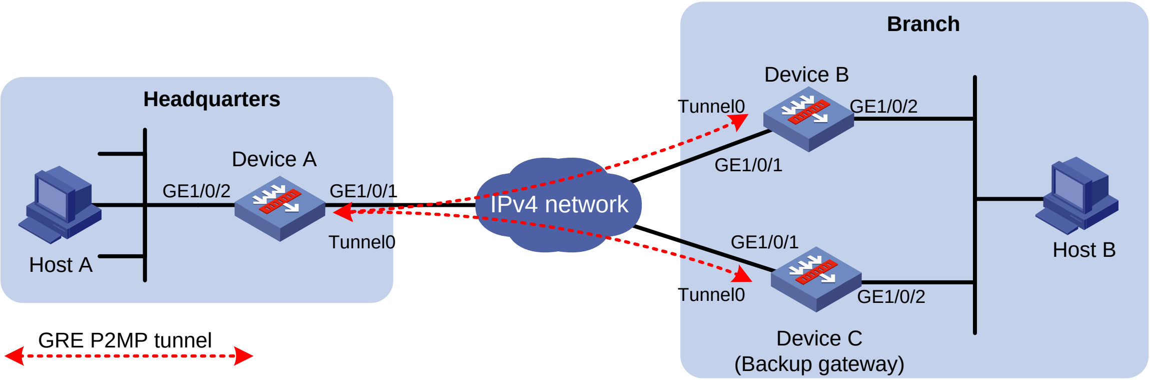


Table 3 IP address assignment

Device	Interface	IP address	Security zone
Device A	GE1/0/1	11.1.1.1/24	Untrust
	GE1/0/2	172.17.17.1/24	Trust
	Tunnel 0	192.168.22.1/24	Untrust
Device B	GE1/0/1	11.1.2.2/24	Untrust
	GE1/0/2	192.168.1.2/24	Trust
	Tunnel 0	192.168.22.2/24	Untrust
Device C	GE1/0/1	11.1.3.3/24	Untrust
	GE1/0/2	192.168.1.3/24	Trust
	Tunnel 0	192.168.22.3/24	Untrust

Procedure

1. Configure Device A:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 11.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ip address 172.17.17.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/2] quit
```

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE P2MP.

```
[DeviceA] interface tunnel 0 mode gre-p2mp
[DeviceA-Tunnel0] ip address 192.168.22.1 255.255.255.0
[DeviceA-Tunnel0] gre p2mp branch-network-mask 255.255.255.0
[DeviceA-Tunnel0] gre p2mp aging-time 20
[DeviceA-Tunnel0] source 11.1.1.1
[DeviceA-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.1.2.

```
[DeviceA] ip route-static 11.1.2.2 24 11.1.1.2
[DeviceA] ip route-static 11.1.3.3 24 11.1.1.2
[DeviceA] ip route-static 192.168.1.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceA] security-zone name Untrust
[DeviceA-security-zone-Untrust] import interface Tunnel 0
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name Trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device A to send packets to Device B and Device C.

```
[DeviceA] security-policy ip
```

```
[DeviceA-security-policy-ip] rule name grelocalout
[DeviceA-security-policy-ip-1-grelocalout] source-zone local
[DeviceA-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 11.1.1.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 192.168.22.1
[DeviceA-security-policy-ip-1-grelocalout] source-ip-host 172.17.17.1
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 11.1.2.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 11.1.3.3
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.3
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.1.2
[DeviceA-security-policy-ip-1-grelocalout] destination-ip-host 192.168.1.3
[DeviceA-security-policy-ip-1-grelocalout] action pass
[DeviceA-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device A to receive the packets sent from Device B and Device C.

```
[DeviceA-security-policy-ip] rule name grelocalin
[DeviceA-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceA-security-policy-ip-2-grelocalin] destination-zone local
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 11.1.2.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 11.1.3.3
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.3
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.1.2
[DeviceA-security-policy-ip-2-grelocalin] source-ip-host 192.168.1.3
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 11.1.1.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 192.168.22.1
[DeviceA-security-policy-ip-2-grelocalin] destination-ip-host 172.17.17.1
[DeviceA-security-policy-ip-2-grelocalin] action pass
[DeviceA-security-policy-ip-2-grelocalin] quit
[DeviceA-security-policy-ip] quit
```

2. Configure Device B:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 11.1.2.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 192.168.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/2] quit
```

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE/IPv4.

```
[DeviceB] interface tunnel 0 mode gre
[DeviceB-Tunnel0] ip address 192.168.22.2 255.255.255.0
[DeviceB-Tunnel0] source 11.1.2.2
[DeviceB-Tunnel0] destination 11.1.1.1
[DeviceB-Tunnel0] gre key 1
[DeviceB-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.2.3.

```
[DeviceB] ip route-static 11.1.1.1 24 11.1.2.3
[DeviceB] ip route-static 172.17.17.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceB] security-zone name Untrust
[DeviceB-security-zone-Untrust] import interface Tunnel 0
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name Trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device B to send packets to Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name grelocalout
[DeviceB-security-policy-ip-1-grelocalout] source-zone local
[DeviceB-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 11.1.2.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 192.168.22.2
[DeviceB-security-policy-ip-1-grelocalout] source-ip-host 192.168.1.2
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 11.1.1.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.1
[DeviceB-security-policy-ip-1-grelocalout] destination-ip-host 172.17.17.1
[DeviceB-security-policy-ip-1-grelocalout] action pass
[DeviceB-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device B to receive the packets sent from Device A.

```
[DeviceB-security-policy-ip] rule name grelocalin
[DeviceB-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceB-security-policy-ip-2-grelocalin] destination-zone local
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 11.1.1.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.1
[DeviceB-security-policy-ip-2-grelocalin] source-ip-host 172.17.17.1
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 11.1.2.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 192.168.22.2
[DeviceB-security-policy-ip-2-grelocalin] destination-ip-host 192.168.1.2
[DeviceB-security-policy-ip-2-grelocalin] action pass
[DeviceB-security-policy-ip-2-grelocalin] quit
[DeviceB-security-policy-ip] quit
```

3. Configure Device C:

# Assign IP addresses to interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ip address 11.1.3.3 255.255.255.0
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] ip address 192.168.1.3 255.255.255.0
```

```
[DeviceC-GigabitEthernet1/0/2] quit
```

# Create tunnel interface **Tunnel 0** and specify the tunnel mode as GRE/IPv4.

```
[DeviceC] interface tunnel 0 mode gre
[DeviceC-Tunnel0] ip address 192.168.22.3 255.255.255.0
[DeviceC-Tunnel0] source 11.1.3.3
[DeviceC-Tunnel0] destination 11.1.1.1
[DeviceC-Tunnel0] gre key 2
[DeviceC-Tunnel0] quit
```

# Configure settings for routing. This example configures static routes. In the routes, the output interface is Tunnel 0 and the next hop is 11.1.3.4.

```
[DeviceC] ip route-static 11.1.1.1 24 11.1.3.4
[DeviceC] ip route-static 172.17.17.0 24 tunnel 0
```

# Add interfaces to security zones.

```
[DeviceC] security-zone name Untrust
[DeviceC-security-zone-Untrust] import interface Tunnel 0
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceC-security-zone-Untrust] quit
[DeviceC] security-zone name Trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
```

# Configure a rule named **grelocalout** in the IPv4 security policy to allow Device C to send packets to Device A.

```
[DeviceC] security-policy ip
[DeviceC-security-policy-ip] rule name grelocalout
[DeviceC-security-policy-ip-1-grelocalout] source-zone local
[DeviceC-security-policy-ip-1-grelocalout] destination-zone untrust
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 11.1.3.3
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 192.168.22.3
[DeviceC-security-policy-ip-1-grelocalout] source-ip-host 192.168.1.3
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 11.1.1.1
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 192.168.22.1
[DeviceC-security-policy-ip-1-grelocalout] destination-ip-host 172.17.17.1
[DeviceC-security-policy-ip-1-grelocalout] action pass
[DeviceC-security-policy-ip-1-grelocalout] quit
```

# Configure a rule named **grelocalin** in the IPv4 security policy to allow Device C to receive the packets sent from Device A.

```
[DeviceC-security-policy-ip] rule name grelocalin
[DeviceC-security-policy-ip-2-grelocalin] source-zone untrust
[DeviceC-security-policy-ip-2-grelocalin] destination-zone local
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 11.1.1.1
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 192.168.22.1
[DeviceC-security-policy-ip-2-grelocalin] source-ip-host 172.17.17.1
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 11.1.3.3
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 192.168.22.3
[DeviceC-security-policy-ip-2-grelocalin] destination-ip-host 192.168.1.3
[DeviceC-security-policy-ip-2-grelocalin] action pass
[DeviceC-security-policy-ip-2-grelocalin] quit
[DeviceC-security-policy-ip] quit
```

Verifying the configuration

# Configure Device C as the default gateway of Host B, and then ping Host A from Host B. The ping operation succeeds. (Details not shown.)

# Display GRE P2MP tunnel entries on Device A.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:1
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.1.0           255.255.255.0   11.1.3.3                2         20
```

# Configure the default gateway of Host B as Device B, and then ping Host A from Host B. The ping operation succeeds. (Details not shown).

# Display GRE P2MP tunnel entries on Device A.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:2
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.1.0           255.255.255.0   11.1.3.3                2         20
192.168.1.0           255.255.255.0   11.1.2.2                1         20
```

The output shows that Device A has two GRE P2MP tunnel entries destined for the branch. The device prefers the tunnel entry with a lower GRE key value. The entry destined for Device B is preferred.

# Shut down tunnel interface **Tunnel 0** on Device B.

```
[DeviceB] interface tunnel 0
[DeviceB-Tunnel0] shutdown
[DeviceB-Tunnel0] quit
```

# Change the default gateway of Host B to Device C. (Details not shown.)

# Ping Host A from Host B after the GRE P2MP tunnel entry destined for Device B ages. The ping operation succeeds. (Details not shown.)

# Display GRE P2MP tunnel entries on Device A.

```
[DeviceA] display gre p2mp tunnel-table interface tunnel 0
Total number:1
Dest Addr             Mask/Prefix Len Tunnel Dest Addr        Gre Key   Aging
192.168.1.0           255.255.255.0   11.1.1.3                2         10
```

The output shows that Device A can forward traffic to hosts in the branch only through Device C.

Contents

Configuring L2TP

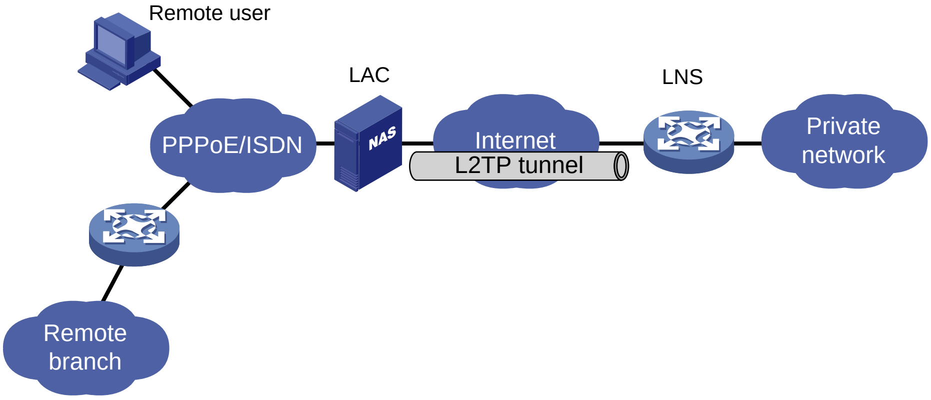
About L2TP

The Layer 2 Tunneling Protocol (L2TP) is a Virtual Private Dialup Network (VPDN) tunneling protocol. L2TP sets up point-to-point tunnels across a public network (for example, the Internet) and transmits encapsulated PPP frames (L2TP packets) over the tunnels. With L2TP, remote users can access the private networks through L2TP tunnels after connecting to a public network by using PPP.

As a Layer 2 VPN technology, L2TP provides a secure, cost-effective solution for remote users to access private networks.

Typical L2TP networking

Figure 1 L2TP network diagram



As shown in Figure 1, a typical L2TP network has the following components:

- **Remote system**—A remote system is usually a remote user's host or a remote branch's device that needs to access the private network.
- **LAC**—An L2TP access concentrator (LAC) is both PPP and L2TP capable. It is usually a network access server (NAS) located at a local ISP, which provides access services mainly for PPP users.

  An LAC is an endpoint of an L2TP tunnel and lies between an LNS and a remote system. It encapsulates packets received from a remote system by using L2TP and then sends the encapsulated packets to the LNS. It decapsulates packets received from the LNS and then sends the decapsulated packets to the intended remote system.
- **LNS**—An L2TP network server (LNS) is both PPP and L2TP capable. It is usually an edge device on an enterprise network.

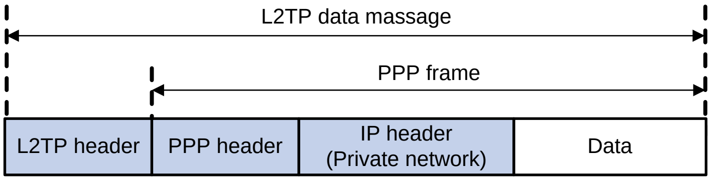
  An LNS is the other endpoint of an L2TP tunnel. It is the logical termination point of a PPP session tunneled by the LAC. L2TP extends the termination point of a PPP session from a NAS to an LNS by establishing a tunnel.

L2TP message types and encapsulation structure

L2TP uses the following types of messages:

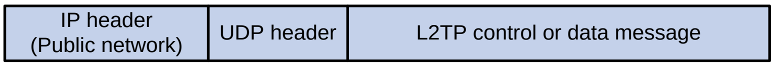
- **Control messages**—Used to establish, maintain, and delete L2TP tunnels and sessions. Control messages are transmitted over a reliable control channel, which supports flow control and congestion control.
- **Data messages**—Used to encapsulate PPP frames, as shown in Figure 2. Data messages are transmitted over an unreliable data channel and are not retransmitted when packet loss occurs. Data messages can use sequence numbers to reorder packets that are disordered during transport.

Figure 2 Data message format



As shown in Figure 3, both control messages and data messages are encapsulated in UDP datagrams.

Figure 3 L2TP encapsulation structure



L2TP tunnel and session

An L2TP tunnel is a virtual point-to-point connection between an LAC and an LNS. Multiple L2TP tunnels can be established between an LNS and an LAC. An L2TP tunnel can carry one or more L2TP sessions. Each L2TP session corresponds to a PPP session and is multiplexed on an L2TP tunnel. An L2TP session is established between the LAC and LNS when an end-to-end PPP session is established between a remote system and the LNS. Data frames for the PPP session are transmitted over the tunnel between the LAC and LNS.

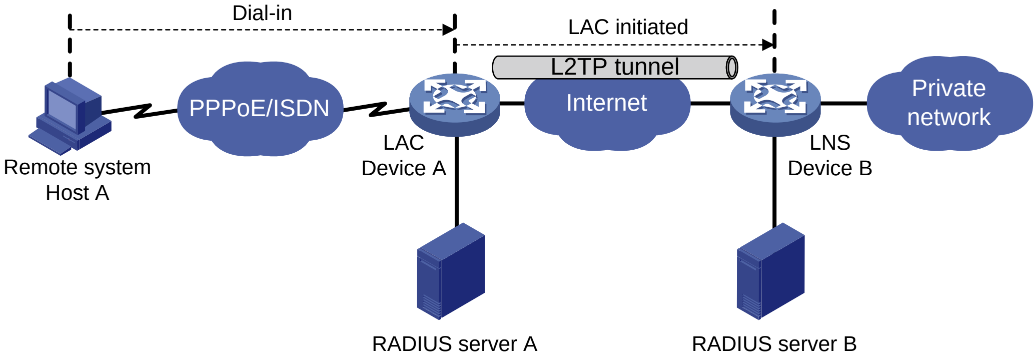
L2TP tunneling modes and tunnel establishment process

L2TP tunneling modes include NAS-initiated, client-initiated, and LAC-auto-initiated.

NAS-initiated tunneling mode

As shown in Figure 4, a remote system dials in to the LAC through a PPPoE network. The LAC initiates a tunneling request to the LNS over the Internet.

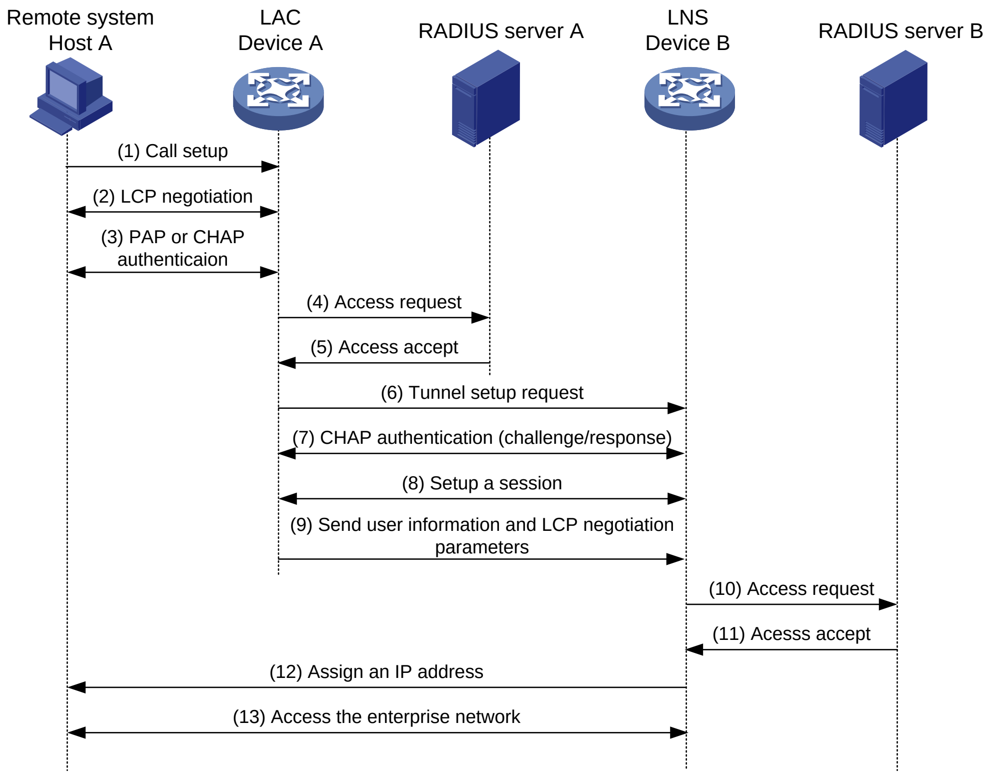
Figure 4 NAS-initiated tunneling mode



A NAS-initiated tunnel has the following characteristics:

- The remote system only needs to support PPP, and it does not need to support L2TP.
- Authentication and accounting of the remote system can be implemented on the LAC or LNS.

Figure 5 NAS-initiated tunnel establishment process



As shown in Figure 5, the following workflow is used to establish a NAS-initiated tunnel:

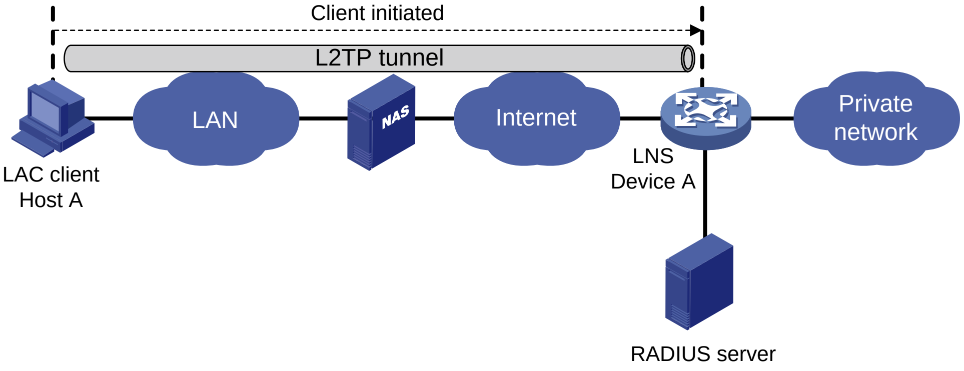
1. A remote system (Host A) initiates a PPP connection to the LAC (Device A).
2. The remote system and LAC perform PPP LCP negotiation.
3. The LAC authenticates PPP user information of Host A by using PAP or CHAP.
4. The LAC sends the authentication information (username and password) to its RADIUS server (RADIUS server A) for authentication.
5. RADIUS server A authenticates the user and returns the result.
6. The LAC initiates an L2TP tunneling request to the LNS (Device B) when the following conditions exist:
   - The user passes the authentication.
   - The user is determined to be an L2TP user according to the username or the ISP domain to which the user belongs.
7. If tunnel authentication is needed, the LAC and LNS send CHAP challenge messages to authenticate each other before successfully establishing an L2TP tunnel.
8. The LAC and LNS negotiate to establish L2TP sessions.
9. The LAC sends PPP user information and PPP negotiation parameters to the LNS.
10. The LNS sends the authentication information to its RADIUS server (RADIUS server B) for authentication.
11. RADIUS server B authenticates the user and returns the result.
12. If the user passes the authentication, the LNS assigns a private IP address to the remote system (Host A).
13. The PPP user can access internal resources of the enterprise.

In steps 12 and 13, the LAC forwards packets for the remote system and LNS. Host A and LAC exchange PPP frames, and the LAC and LNS exchange L2TP packets.

Client-initiated tunneling mode

As shown in Figure 6, a remote system running L2TP (LAC client) has a public IP address to communicate with the LNS through the Internet. The LAC client can directly initiate a tunneling request to the LNS without any dedicated LAC devices.

Figure 6 Client-initiated tunneling mode

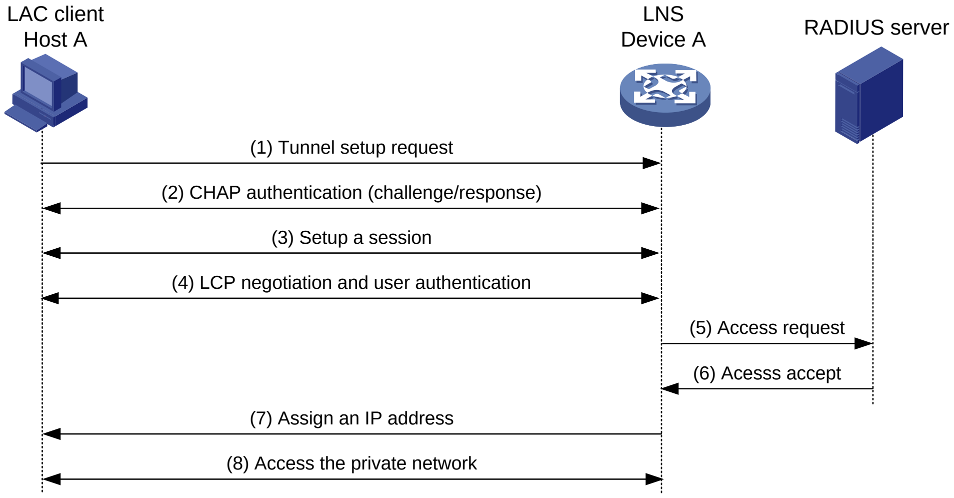


A client-initiated tunnel has the following characteristics:

- A client-initiated tunnel has higher security because it is established between a remote system and the LNS.
- The remote system must support L2TP and be able to communicate with the LNS. This causes poor expandability.

As shown in Figure 7, the workflow for establishing a client-initiated tunnel is similar to that for establishing a NAS-initiated tunnel. (Details not shown.)

Figure 7 Client-initiated tunnel establishment process

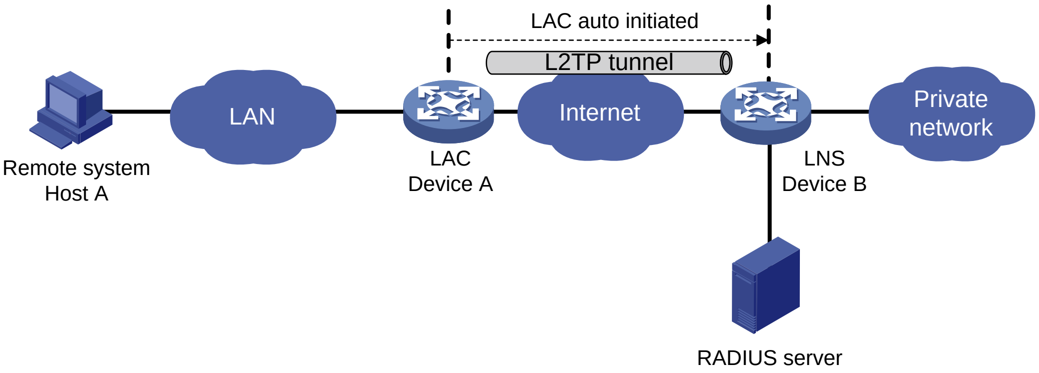


LAC-auto-initiated tunneling mode

In NAS-initiated mode, a remote system must successfully dial in to the LAC through PPPoE.

In LAC-auto-initiated mode, you can use the **`l2tp-auto-client`** command on the LAC to trigger the LAC to initiate a tunneling request to the LNS. When a remote system accesses the private network, the LAC forwards data through the L2TP tunnel.

Figure 8 LAC-auto-initiated tunneling mode

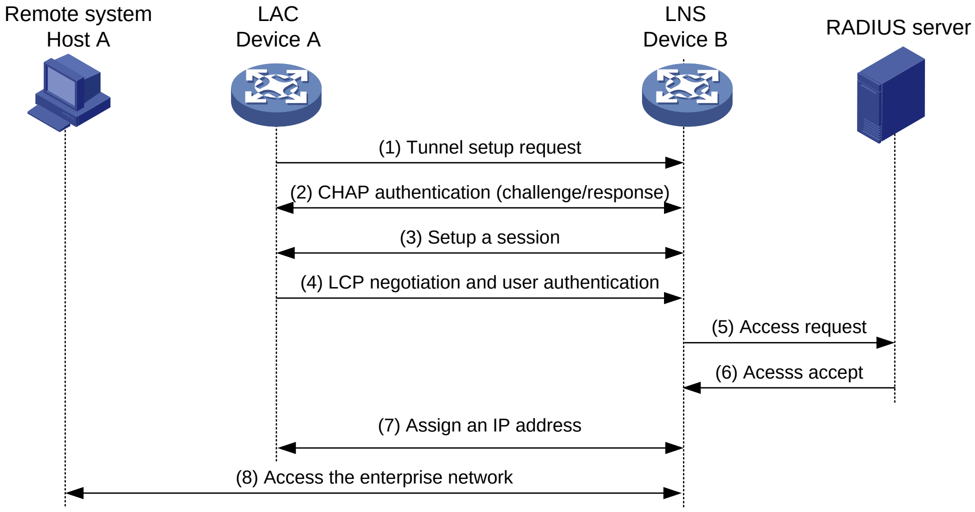


An LAC-auto-initiated tunnel has the following characteristics:

- The connection between a remote system and the LAC is not confined to a dial-up connection and can be any IP-based connection.
- An L2TP session is established immediately after an L2TP tunnel is established. Then, the LAC and LNS, acting as the PPPoE client and PPPoE server, respectively, perform PPP negotiation.
- The LNS assigns a private IP address to the LAC instead of to the remote system.

As shown in Figure 9, the workflow for establishing an LAC-auto-initiated tunnel is similar to that for establishing a NAS-initiated tunnel. (Details not shown.)

Figure 9 Establishment process for LAC-auto-initiated tunnels



L2TP features

- **Flexible identity authentication mechanism and high security**—L2TP by itself does not provide security for connections. However, it has all the security features of PPP and allows for PPP authentication (CHAP or PAP). L2TP can also cooperate with IPsec to improve security for tunneled data.
- **Multiprotocol transmission**—L2TP tunnels PPP frames, which can be used to encapsulate packets of multiple network layer protocols.
- **RADIUS authentication**—An LAC or LNS can send the username and password of a remote user to a RADIUS server for authentication.

- **Private address allocation**—An LNS can dynamically allocate private addresses to remote users. This facilitates address allocation for private internets (RFC 1918) and improves security.
- **Flexible accounting**—Accounting can be simultaneously performed on the LAC and LNS. This allows bills to be generated on the ISP side and charging and auditing to be processed on the enterprise gateway. L2TP can provide accounting data, including inbound and outbound traffic statistics (in packets and bytes) and the connection's start time and end time. The AAA server uses these data for flexible accounting.
- **Reliability**—L2TP supports LNS backup. When the connection to the primary LNS is torn down, an LAC can establish a new connection to a secondary LNS. This redundancy enhances the reliability of L2TP services.
- **Issuing tunnel attributes by RADIUS server to LAC**—In NAS-initiated mode, the tunnel attributes can be issued by the RADIUS server to the LAC. For the LAC to receive these attributes, enable L2TP and configure remote AAA authentication for PPP users on the LAC.

  When an L2TP user dials in to the LAC, the LAC as the RADIUS client sends the user information to the RADIUS server. The RADIUS server authenticates the PPP user, returns the result to the LAC, and issues L2TP tunnel attributes for the PPP user to the LAC. The LAC then sets up an L2TP tunnel and sessions based on the issued L2TP tunnel attributes.

  Table 1 Tunnel attributes that can be issued by the RADIUS server

Attribute number	Attribute name	Description
64	Tunnel-Type	Tunnel type, which can only be L2TP.
65	Tunnel-Medium-Type	Transmission medium type for the tunnel, which can only be IPv4.
67	Tunnel-Server-Endpoint	IP address of the LNS.
69	Tunnel-Password	Key used to authenticate a peer of the tunnel.
81	Tunnel-Private-Group-ID	Group ID for the tunnel. The LAC sends this value to the LNS for the LNS to perform an operation accordingly.
82	Tunnel-Assignment-ID	Assignment ID for the tunnel. It is used to indicate the tunnel to which a session is assigned. L2TP users with the same Tunnel-Assignment-ID, Tunnel-Server-Endpoint, and Tunnel-Password attributes share an L2TP tunnel.
90	Tunnel-Client-Auth-ID	Tunnel name. It is used to indicate the local tunnel.

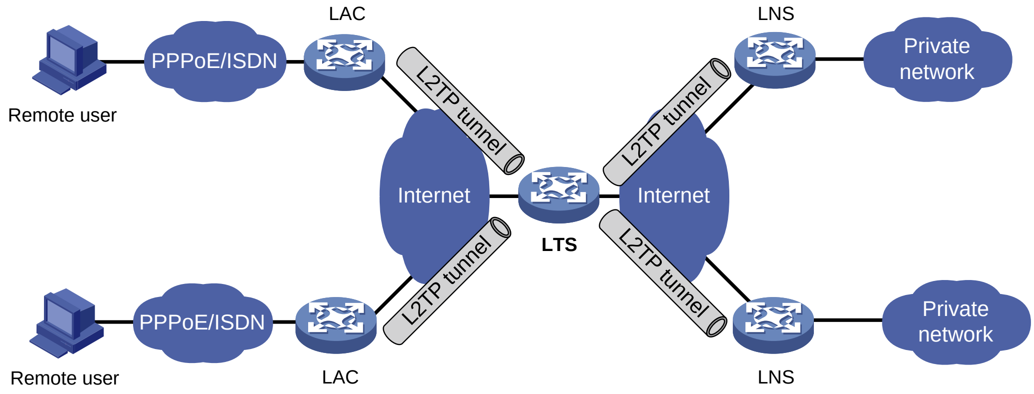
  The RADIUS server can issue only one set of the L2TP tunnel attributes in a RADIUS packet.

  The RADIUS-issued tunnel attributes override the tunnel attributes manually configured on the LAC, but not vice versa.
- **L2TP tunnel switching**—Also called multihop L2TP tunneling. As shown in Figure 10, the Layer 2 tunnel switch (LTS) terminates L2TP packets from each LAC as an LNS. It then sends these packets to a destination LNS as an LAC.

  L2TP tunnel switching has the following features:
  - **Simplified configuration and deployment**—When LACs and LNSs are in different management domains:
    - All LACs consider the LTS as an LNS and do not need to differentiate LNSs on the network.
    - All LNSs consider the LTS as an LAC and are not affected by the addition or deletion of LACs.

- **L2TP tunnel sharing**—Different users can share the same L2TP tunnel between the LAC and the LTS. The LTS distributes data of different users to different LNSs.

Figure 10 L2TP tunnel switching network diagram



L2TP-based EAD

EAD authenticates PPP users that pass the access authentication. PPP users that pass EAD authentication can access network resources. PPP users that fail EAD authentication can only access the resources in the quarantine areas.

EAD uses the following procedure:

1. The iNode client uses L2TP to access the LNS. After the client passes the PPP authentication, the CAMS/IMC server assigns isolation ACLs to the LNS. The LNS uses the isolation ACLs to filter incoming packets.
2. After the IPCP negotiation, the LNS sends the IP address of the CAMS/IMC server to the iNode client. The server IP address is permitted by the isolation ACLs.
3. The CAMS/IMC server authenticates the iNode client and performs security check for the iNode client. If the iNode client passes security check, the CAMS/IMC server assigns security ACLs for the iNode client to the LNS. The iNode client can access network resources.

Protocols and standards

- RFC 1661, *The Point-to-Point Protocol (PPP)*
- RFC 1918, *Address Allocation for Private Internets*
- RFC 2661, *Layer Two Tunneling Protocol "L2TP"*
- RFC 2868, *RADIUS Attributes for Tunnel Protocol Support*

Prerequisites for L2TP

When you configure L2TP, perform the following tasks:

1. Determine the network devices needed according to the networking environment.
   - For NAS-initiated mode and LAC-auto-initiated mode, configure both the LAC and the LNS.
   - For client-initiated mode, you only need to configure the LNS.
2. Configure the devices based on the intended role (LAC or LNS) on the network.

L2TP tasks at a glance

LAC tasks at a glance

To configure a device as an LAC in NAS-initiated or LAC-auto-initiated mode, complete the following tasks:

1. Configuring basic L2TP capabilities
2. Configuring an LAC
   - Configuring an LAC to initiate tunneling requests for a user

     This task is required for NAS-initiated mode and unnecessary for LAC-auto-initiated mode.
   - Specifying LNS IP addresses
   - Configuring the source IP address of L2TP tunnel packets
   - (Optional) Enabling transferring AVP data in hidden mode
   - Configuring AAA authentication on an LAC

     This task is required for NAS-initiated mode and unnecessary for LAC-auto-initiated mode.
   - Configuring an LAC to automatically establish an L2TP tunnel

     This task is required for NAS-initiated mode and unnecessary for LAC-auto-initiated mode.
   - (Optional) Configuring the polling feature
   - (Optional) Restoring the default settings for a virtual PPP interface
3. (Optional) Configuring optional L2TP parameters
   - Configuring L2TP tunnel authentication
   - Setting the Hello interval
   - Enabling session flow control
   - Setting the DSCP value of L2TP packets
   - Assigning a tunnel peer to a VPN
   - Setting the TSA ID of the LTS

LNS tasks at a glance

1. Configuring basic L2TP capabilities
2. Configuring an LNS
   - Creating a VT interface
   - (Optional.) Configuring a static VA pool
   - Configuring an LNS to accept L2TP tunneling requests from an LAC
   - (Optional.) Configuring user authentication on an LNS
   - (Optional.) Configuring AAA authentication on an LNS
   - (Optional.) Setting the maximum number of ICRQ packets that the LNS can process per second
   - (Optional.) Logging out an old L2TP user when the IP addresses of the old user and new user conflict
3. (Optional) Configuring optional L2TP parameters
   - Configuring L2TP tunnel authentication
   - Setting the Hello interval
   - Enabling session flow control

Configuring basic L2TP capabilities

About this task

Basic L2TP capability configuration includes the following tasks:

- **Enabling L2TP**—L2TP must be enabled for L2TP configurations to take effect.
- **Creating an L2TP group**—An L2TP group is intended to represent a group of parameters. This enables not only flexible L2TP configuration on devices, but also one-to-one and one-to-many networking applications for LACs and LNSs. An L2TP group has local significance only. However, the relevant settings of the L2TP groups on the LAC and LNS must match. For example, the local tunnel name configured on the LAC must match the tunnel peer name configured on the LNS.
- **Configuring the local tunnel name**—The local tunnel name identifies the tunnel at the local end during tunnel negotiation between an LAC and an LNS.

Procedure

1. Enter system view.

   `system-view`

2. Enable L2TP.

   `l2tp enable`

   By default, L2TP is disabled.

3. Create an L2TP group, specify its mode, and enter its view.

   `l2tp-group` *`group-number`* **`mode`** { **`lac`** | **`lns`** }

   Specify the mode as **`lac`** on the LAC side and as **`lns`** on the LNS side.

4. Specify the local tunnel name.

   `tunnel name` *`name`*

   By default, the device name is used.

   The local tunnel name configured on the LAC must match the tunnel peer name configured on the LNS.

Configuring an LAC

Configuring an LAC to initiate tunneling requests for a user

About this task

This task configures an LAC to initiate tunneling requests to an LNS for a user. When the PPP user information matches the specified user, the LAC determines that the PPP user is an L2TP user and initiates tunneling requests to the LNS.

You can specify a user by configuring one of the following items:

- **Fully qualified name**—The LAC initiates tunneling requests to the LNS only if the username of a PPP user matches the configured fully qualified name.

- **Domain name**—The LAC initiates tunneling requests to the LNS only if the ISP domain name of a PPP user matches the configured domain name.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view in LAC mode.

   l2tp-group *group-number* [**mode lac**]

3. Configure the LAC to initiate tunneling requests for a user.

   user { **domain** *domain-name* | **fullusername** *user-name* }

   By default, an LAC does not initiate tunneling requests for any users.

Specifying LNS IP addresses

About this task

You can specify up to five LNS IP addresses and domain names (the total number of IP addresses and domain names cannot exceed five). The LAC initiates an L2TP tunneling request to its specified LNSs consecutively in their configuration order until it receives an acknowledgment from an LNS. That LNS then becomes the tunnel peer.

When the IP address of an LNS is fixed, you can specify the LNS IP address by using the **lns-ip** *ip-address* command. When the IP address of an LNS is not fixed, you can specify the LNS domain name by using the **lns-ip host-name** command. In this case, the LAC will deliver the domain name to the DNS module for processing. Then, the LAC can initiate an L2TP tunneling request to the LNS according to the returned IP address. For more information about DNS, see *Layer 3—IP Services Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view in LAC mode.

   l2tp-group *group-number* [**mode lac**]

3. Specify LNS IP addresses or domain names.

   lns-ip { *ip-address* | **host-name** *name* }&<1-5>

   By default, no LNS IP addresses or domain names are specified.

Configuring the source IP address of L2TP tunnel packets

Restrictions and guidelines

For high availability, as a best practice, use the IP address of a loopback interface as the source IP address of L2TP tunnel packets on the LAC. If equal cost routing paths exist between the LAC and LNS, you must use the IP address of a loopback interface as the source IP address of L2TP tunnel packets. To do so, use the **source-ip** command or use the RADIUS server to assign a loopback interface address.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view in LAC mode.

   l2tp-group *group-number* [**mode lac**]

3. Configure the source IP address of L2TP tunnel packets.

   ```
   source-ip ip-address
   ```

   By default, the source IP address of L2TP tunnel packets is the IP address of the egress interface.

Enabling transferring AVP data in hidden mode

About this task

L2TP uses Attribute Value Pairs (AVPs) to transmit tunnel negotiation parameters, session negotiation parameters, and user authentication information. Transferring AVP data in hidden mode can hide sensitive AVP data such as user passwords. This feature encrypts AVP data with the key configured by using the **tunnel password** command before transmission.

Restrictions and guidelines

This configuration takes effect only when the tunnel authentication feature is enabled. For more information about configuring tunnel authentication, see "Configuring L2TP tunnel authentication."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter L2TP group view in LAC mode.

   ```
   l2tp-group group-number [ mode lac ]
   ```

3. Enable transferring AVP data in hidden mode.

   ```
   tunnel avp-hidden
   ```

   By default, AVP data is transferred in plain text.

Configuring AAA authentication on an LAC

You can configure AAA authentication on an LAC to authenticate the remote dialup users and initiate a tunneling request only for qualified users. A tunnel will not be established for unqualified users.

The device supports both local AAA authentication and remote AAA authentication.

- For local AAA authentication, create a local user and configure a password for each remote user on the LAC. The LAC then authenticates a remote user by matching the provided username and password with those configured locally.
- For remote AAA authentication, configure the username and password of each user on the RADIUS/HWTACACS server. The LAC then sends the remote user's username and password to the server for authentication.

For more information about AAA, see *Security Configuration Guide*.

To enable AAA authentication on an LAC, you also need to configure PAP or CHAP authentication for PPP users on the user access interfaces. For information about configuring PAP or CHAP, see Configuring PPP in *Layer 2—WAN Access Configuration Guide.*

Configuring an LAC to automatically establish an L2TP tunnel

1. Enter system view.

   ```
   system-view
   ```

2. Create a virtual PPP interface and enter its view.

interface virtual-ppp *interface-number*

3. Configure the IP address of the virtual PPP interface.
   - Assign an IP address to the virtual PPP interface.

     ip address *address mask*

     By default, no IP address is configured.
   - Enable IP address negotiation on the virtual PPP interface.

     ip address ppp-negotiate

     By default, IP address negotiation on a virtual PPP interface is disabled.
4. Configure the peer to be authenticated.

   Use the **ppp pap** or **ppp chap** command to specify the PPP authentication method and configure the username and password of the PPP user. The LNS then authenticates the PPP user. For more information, see PPP commands in *Layer 2—WAN Access Command Reference.*
5. (Optional.) Set the description for the interface.

   description *text*

   By default, the description of an interface is in the format of *interface-name* **Interface**, for example, **Virtual-PPP254 Interface**.
6. (Optional.) Set the MTU size of the interface.

   mtu *size*

   The default setting is 1500 bytes.
7. (Optional.) Set the expected bandwidth for the interface.

   bandwidth *bandwidth-value*

   By default, the expected bandwidth (in kbps) is interface baudrate divided by 1000.
8. (Optional.) Bring up the interface.

   undo shutdown

   By default, an interface is up.
9. Configure the LAC to automatically establish an L2TP tunnel with the LNS.

   l2tp-auto-client l2tp-group *group-number*

   By default, an LAC does not establish an L2TP tunnel.

   An L2TP tunnel automatically established in LAC-auto-initiated mode exists until you remove the tunnel by using the **undo l2tp-auto-client** or **undo l2tp-group** *group-number* command.

Configuring the polling feature

About this task

The polling feature checks L2TP link state.

On an interface that uses L2TP encapsulation, the link layer sends keepalives at keepalive intervals to detect the availability of the peer. If the interface fails to receive keepalives when the keepalive retry limit is reached, it tears down the link and reports a link layer down event.

To set the keepalive retry limit, use the **timer-hold retry** command.

The keepalive interval of 0 disables sending of keepalives.

Restrictions and guidelines

On a slow link, increase the keepalive interval to prevent false shutdown of the interface. This situation might occur when keepalives are delayed because a large packet is being transmitted on the link.

The keepalive interval must be smaller than the negotiation timeout time.

Procedure

1. Enter system view.
   system-view
2. Enter virtual PPP interface view.
   interface virtual-ppp *interface-number*
3. Set the keepalive interval.
   timer-hold *seconds*
   The default setting is 10 seconds.
4. Set the keepalive retry limit.
   timer-hold retry *retries*
   The default setting is 5.

Restoring the default settings for a virtual PPP interface

Restrictions and guidelines

The **default** command might interrupt ongoing network services. Make sure you are fully aware of the impact of this command when you execute it on a live network.

The **default** command might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **display this** command in interface view to identify these commands. Use the **undo** forms of these commands or follow the command reference to individually restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.
   system-view
2. Enter virtual PPP interface view.
   interface virtual-ppp *interface-number*
3. Restore the default settings for the interface.
   default

Configuring an LNS

Creating a VT interface

After an L2TP session is established, a virtual access (VA) interface is needed for data exchange with the peer. The system will dynamically create VA interfaces based on the parameters of the virtual template (VT) interface. To configure an LNS, first create a VT interface and configure the following parameters for it:

- Interface IP address.
- Authentication mode for PPP users.

- IP addresses allocated by the LNS to PPP users.

For information about configuring VT interfaces, see PPP configuration in *Layer 2—WAN Access Configuration Guide* and IP addressing configuration in *Layer 3—IP Services Configuration Guide*.

Configuring a static VA pool

About this task

A VA pool contains a group of VA interfaces. You can configure a VA pool to improve the performance of establishing or terminating L2TP connections. The LNS selects a VA interface from the pool for a requesting user and releases the VA interface when the user goes offline.

Restrictions and guidelines

On a device that does not support dynamic VA pools:

When you configure a static VA pool, follow these guidelines:

- A VT interface can be associated with only one static VA pool. To change the capacity of a static VA pool, delete the previous configuration, and reconfigure the static VA pool.
- Creating or deleting a static VA pool takes time. During the process of creating or deleting a static VA pool, users can come online or go offline, but the static VA pool does not take effect.
- The system might create a static VA pool that contains VA interfaces less than the specified number because of insufficient resources. In this case, you can use the **`display l2tp va-pool`** command to view the number of available VA interfaces and current state of the static VA pool.
- Create a static VA pool with an appropriate capacity, because a static VA pool occupies much system memory.
- Deleting a static VA pool does not log off the users who are using VA interfaces in the static VA pool.

On a device that supports dynamic VA pools:

L2TP supports the following types of VA pools:

- **Static VA pool**—VA pool manually created by using the **`l2tp virtual-template va-pool`** command.
- **Dynamic VA pool**—VA pool automatically created by the device.

When an L2TP user comes online, the device select a VA interface for the user in the following order:

1. VA interfaces in the static VA pool.
2. VA interfaces in the dynamic VA pool.

If no static VA pool is configured for a VT interface or the static VA pool configured for a VT interface is exhausted, the following rules apply when a new L2TP user comes online:

- If no dynamic VA pool is created for the VT interface, the device first creates a dynamic VA pool containing 128 VA interfaces for the VT interface. Then, the device allocates a VA interface in the dynamic VA pool to the user.
- If a dynamic VA pool with more than 64 available VA interfaces exists for the VT interface, the device will allocate a VA interface in the dynamic VA pool to the user.
- If a dynamic VA pool with less than 64 available VA interfaces exists for the VT interface, the device adds 128 VA interfaces to the dynamic VA pool. Then, the device allocates a VA interface in the dynamic VA pool to the user.

The VA pool occupies certain memory resources. When the device memory is large or the user scale is stable, as a best practice, create a static VA pool of a suitable capacity. When the device memory is small or the user scale is uncertain, as a best practice, use a dynamic VA pool. In this case, the device can automatically create a dynamic VA pool with the number of VA interfaces at the step of 128 according to the network user scale.

For a VA pool, follow these restrictions and guidelines:

- Static VA pool
  - A VT interface can be associated with only one static VA pool. To change the capacity of a static VA pool, delete the previous configuration, and reconfigure the static VA pool.
  - Creating or deleting a static VA pool takes time. During the process of creating or deleting a static VA pool, users can come online or go offline, but the static VA pool does not take effect.
  - The system might create a static VA pool that contains VA interfaces less than the specified number because of insufficient resources. In this case, you can use the **display l2tp va-pool** command to view the number of available VA interfaces and current state of the static VA pool.
  - Create a static VA pool with an appropriate capacity, because a static VA pool occupies much system memory.
  - Deleting a static VA pool does not log off the users who are using VA interfaces in the static VA pool.
- Dynamic VA pool
  - A dynamic VA pool is automatically created by the device. It cannot be manually configured, modified, or deleted.
  - The device automatically deletes VA interfaces that are not used for a long period of time from the dynamic VA pool to release the memory resources.

Procedure

1. Enter system view.
   system-view
2. Create a static VA pool.
   l2tp virtual-template *template-number* **va-pool** *va-volume*
   By default, no VA pool exists.

Configuring an LNS to accept L2TP tunneling requests from an LAC

About this task

When receiving a tunneling request, an LNS performs the following operations:

- Determines whether to accept the tunneling request by checking whether the name of the tunnel peer (LAC) matches the one configured.
- Determines the VT interface to be used for creating the VA interface.

Procedure

1. Enter system view.
   system-view
2. Enter L2TP group view in LNS mode.
   l2tp-group *group-number* [**mode lns**]
3. Configure the LNS to accept tunneling requests from an LAC and specify the VT interface to be used for tunnel setup.
   - If the L2TP group number is 1:

allow l2tp virtual-template *virtual-template-number* [**remote** *remote-name*]

- If the L2TP group number is not 1:

  allow l2tp virtual-template *virtual-template-number* **remote** *remote-name*

By default, an LNS denies tunneling requests from any LAC.

If the L2TP group number is 1, the **remote** *remote-name* option is optional. If you do not specify this option, the LNS accepts tunneling requests from any LAC.

Configuring user authentication on an LNS

About this task

An LNS can be configured to authenticate a user that has passed authentication on the LAC to increase security. In this case, the user is authenticated once on the LAC and once on the LNS. An L2TP tunnel can be established only when both authentications succeed.

An LNS provides the following authentication methods in ascending order of priority:

- **Proxy authentication**—The LNS uses the LAC as an authentication proxy. The LAC sends the LNS all user authentication information from users and the authentication method configured on the LAC itself. The LNS then checks the user validity according to the received information and the locally configured authentication method.
- **Mandatory CHAP authentication**—The LNS uses CHAP authentication to reauthenticate users who have passed authentication on the LAC.
- **LCP renegotiation**—The LNS ignores the LAC proxy authentication information and performs a new round of LCP negotiation with the user.

The LNS chooses an authentication method depending on your configuration.

- If you configure both LCP renegotiation and mandatory CHAP authentication, the LNS uses LCP renegotiation.
- If you configure only mandatory CHAP authentication, the LNS performs CHAP authentication for users after proxy authentication succeeds.
- If you configure neither LCP renegotiation nor mandatory CHAP authentication, the LNS uses the LAC for proxy authentication.

Restrictions and guidelines for user authentication on an LNS

This mandatory CHAP authentication and LCP renegotiation methods are effective only on NAS-initiated L2TP tunnels.

For mandatory CHAP authentication to take effect, you must also configure CHAP authentication for the PPP user on the VT interface of the LNS.

For the LNS not to accept LCP negotiation parameters, configure this feature to perform a new round of LCP negotiation between the LNS and the user. In this case, the LNS authenticates the user by using the authentication method configured on the corresponding VT interface.

Configuring mandatory CHAP authentication

1. Enter system view.

   system-view

2. Enter L2TP group view in LNS mode.

   l2tp-group *group-number* [**mode lns**]

3. Configure mandatory CHAP authentication.

   mandatory-chap

   By default, CHAP authentication is not performed on an LNS.

Some users might not support the authentication on the LNS. In this situation, do not enable this feature, because CHAP authentication on the LNS will fail.

4. Return to system view.

   ```
   quit
   ```

5. Enter VT interface view and set the authentication type of PPP users to CHAP.

   For more information about VT interfaces, see PPP configuration in *Layer 2—WAN Access Configuration Guide.*

Configuring LCP renegotiation

1. Enter system view.

   ```
   system-view
   ```

2. Enter L2TP group view in LNS mode.

   ```
   l2tp-group group-number [ mode lns ]
   ```

3. Configure the LNS to perform LCP renegotiation with users.

   ```
   mandatory-lcp
   ```

   By default, an LNS does not perform LCP renegotiation with users.

   This command is effective only on NAS-initiated L2TP tunnels.

   If you enable LCP renegotiation but configure no authentication for the corresponding VT interface, the LNS does not perform an additional authentication for users.

Configuring AAA authentication on an LNS

After you configure AAA authentication on an LNS, the LNS can authenticate the usernames and passwords of remote access users. If a user passes AAA authentication, the user can communicate with the LNS to access the private network.

Configure AAA authentication on the LNS in one of the following cases:

- LCP renegotiation is not configured in NAS-initiated mode.
- The VT interface is configured with PPP user authentication and LCP renegotiation is configured in NAS-initiated mode.
- The VT interface is configured with PPP user authentication in client-initiated mode or LAC-auto-initiated mode.

LNS side AAA configurations are similar to those on an LAC (see "Configuring AAA authentication on an LAC").

Setting the maximum number of ICRQ packets that the LNS can process per second

Restrictions and guidelines

To avoid device performance degradation and make sure the LNS can processes ICRQ requests correctly, use this feature to adjust the ICRQ packet processing rate limit.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the maximum number of ICRQ packets that the LNS can process per second.

   ```
   l2tp icrq-limit number
   ```

By default, the maximum number of ICRQ packets that the LNS can process per second is not limited.

Logging out an old L2TP user when the IP addresses of the old user and new user conflict

About this task

When the IP addresses of a new L2TP user and an old L2TP user conflict, you can select to forbid the new user from coming online or log out the old user.

Procedure

1. Enter system view.

   `system-view`

2. Allow a new L2TP user to come online and log out an old L2TP user when the IP addresses of the two user conflict.

   `l2tp user-ip-conflict offline`

   By default, a new L2TP user cannot come online and an old L2TP user keeps online when the IP addresses of the two user conflict.

Configuring optional L2TP parameters

Configuring L2TP tunnel authentication

About this task

Tunnel authentication allows the LAC and LNS to authenticate each other. Either the LAC or the LNS can initiate a tunnel authentication request.

You can enable tunnel authentication on both sides or either side.

To ensure a successful tunnel establishment when tunnel authentication is enabled on both sides or either side, set the same non-null key on the LAC and the LNS. To set the tunnel authentication key, use the **`tunnel password`** command.

When neither side is enabled with tunnel authentication, the key settings of the LAC and the LNS do not affect the tunnel establishment.

Restrictions and guidelines

To ensure tunnel security, enable tunnel authentication.

Modifying the tunnel authentication key does not affect the normal communication of current tunnels. The tunnel authentication key change takes effect at next tunnel establishment.

Procedure

1. Enter system view.

   `system-view`

2. Enter L2TP group view.

   `l2tp-group` *`group-number`* `[` **`mode`** `{` **`lac`** `|` **`lns`** `} ]`

3. Enable L2TP tunnel authentication.

   `tunnel authentication`

   By default, L2TP tunnel authentication is enabled.

4. Set the tunnel authentication key.

tunnel password { **cipher** | **simple** } *string*

By default, no key is set.

Setting the Hello interval

About this task

To check the connectivity of a tunnel, the LAC and LNS periodically send each other Hello packets. At receipt of a Hello packet, the LAC or LNS returns a response packet. If the LAC or LNS receives no response packets from the peer within the Hello interval, it retransmits the Hello packet. If it receives no response packets from the peer after transmitting the Hello packet five times, it considers the L2TP tunnel to be down.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view.

   l2tp-group *group-number* [**mode** { **lac** | **lns** }]

3. Set the Hello interval.

   tunnel timer hello *hello-interval*

   The default setting is 60 seconds.

Enabling session flow control

About this task

This feature adds sequence numbers to transmitted packets and uses them to reorder packets arriving out of order and to detect lost packets.

This feature takes effect on both sent and received L2TP data messages. The L2TP sessions support this feature if either the LAC or LNS is enabled with this feature.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view.

   l2tp-group *group-number* [**mode** { **lac** | **lns** }]

3. Enable the session flow control feature.

   tunnel flow-control

   By default, this feature is disabled.

Setting the DSCP value of L2TP packets

About this task

The DSCP field is the first 6 bits of the IP ToS byte. This field marks the priority of IP packets for forwarding. This feature sets the DSCP value for the IP packet when L2TP encapsulates a PPP frame into an IP packet.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view.

   l2tp-group *group-number* [**mode** { **lac** | **lns** }]

3. Set the DSCP value of L2TP packets.

   ip dscp *dscp-value*

   The default setting is 0.

Assigning a tunnel peer to a VPN

About this task

By default, the device transmits L2TP control messages and data messages over the public network. With this feature, the device transmits them in a VPN by searching the routing table in the VPN.

Restrictions and guidelines

When one L2TP endpoint is in a VPN, assign the peer endpoint to the VPN for correct packet forwarding between the two endpoints.

The tunnel peer and the physical port connecting to the tunnel peer must belong to the same VPN. The VPN to which this physical port belongs is configured by using the **ip binding vpn-instance** command.

Procedure

1. Enter system view.

   system-view

2. Enter L2TP group view.

   l2tp-group *group-number* [**mode** { **lac** | **lns** }]

3. Assign the tunnel peer to a VPN.

   vpn-instance *vpn-instance-name*

   By default, a tunnel peer belongs to the public network.

Setting the TSA ID of the LTS

About this task

To detect loops, the LTS compares the configured TSA ID with each TSA ID AVP in a received ICRQ packet.

- If a match is found, a loop exists. The LTS immediately tears down the session.
- If no match is found, the LTS performs the following operations:
  - Encapsulates the configured TSA ID into a new TSA ID AVP.
  - Appends it to the packet.
  - Sends the packet to the next hop LTS.

Restrictions and guidelines

To avoid loop detection errors, make sure the TSA ID of each LTS is unique.

Procedure

1. Enter system view.

   system-view

2. Set the TSA ID of the LTS and enable L2TP loop detection on the LTS.

   l2tp tsa-id *tsa-id*

By default, the TSA ID of the LTS is not configured, and L2TP loop detection is disabled on the LTS.

Setting the receiving window size for an L2TP tunnel

About this task

To enable the device to process a larger number of disordered packets, use this command to enlarge the receiving window size for an L2TP tunnel.

The device uses a receiving window to reorder disordered packets based on packet sequence numbers.

If the sequence number of a packet is within the receiving window but does not equal the minimum value of the window, the device performs the following operations:

1. The device buffers the packet.
2. The minimum value and maximum value of the receiving window increment by one.
3. The device continues to check the next arriving packet.

If the sequence number of a packet equals the minimum value of the receiving window, the device performs the following operations:

1. The device processes the packet.
2. The minimum value and maximum value of the receiving window increment by one.
3. The device checks buffered packets for a packet with the sequence number equal to the new minimum value of the receiving window.
4. If no required packet is found, the device checks the next arriving packet.

If the sequence number of a packet is not within the receiving window, the device drops the packet.

Restrictions and guidelines

In the L2TP tunnel establishment process, the device uses the value specified in L2TP group view as the receiving window size.

Changing the receiving window size after an L2TP tunnel is established does not affect the established L2TP tunnel.

Procedure

1. Enter system view.

   `system-view`

2. Enter L2TP group view.

   `l2tp-group` *`group-number`* [**mode** { **lac** | **lns** }]

3. Set the receiving window size for the L2TP group.

   `tunnel window receive` *`size`*

   By default, the receiving window size for an L2TP tunnel is 1024.

Setting the sending window size for an L2TP tunnel

About this task

The packet processing capability of a peer end might mismatch the receiving window size of the peer end in some networks. For example, the actual packet processing capability of the peer end is 10, but the receiving window size of the peer end is 20. To ensure stable L2TP services, you can adjust the sending window size for the device to match the actual packet processing capability of the peer end.

Restrictions and guidelines

The sending window size set in L2TP group view is obtained in the L2TP tunnel establishment process.

- If the sending window size is 0, the device uses the default sending window size.
- If the sending window size is not 0, the device uses the specified value as the sending window size.

Changing the sending window size after an L2TP tunnel is established does not affect the established L2TP tunnel.

Procedure

1. Enter system view.

   `system-view`

2. Enter L2TP group view.

   `l2tp-group` *`group-number`* [**`mode`** { **`lac`** | **`lns`** }]

3. Set the sending window size for the L2TP group.

   `tunnel window send` *`size`*

   By default, the sending window size for an L2TP tunnel is 0, which means using the value of the receiving window size carried in messages sent by the peer end in the tunnel establishment process. If the messages from the peer end carry no receiving window size in the tunnel establishment process, the sending window size for the device is 4.

Enabling L2TP-based EAD

About this task

In some network environments that require high security, configure this feature and use the security policy server to perform further security check for users that pass the L2TP authentication. Users that pass EAD authentication can access network resources. Users that fail EAD authentication can only access the resources in the quarantine areas.

Restrictions and guidelines

EAD authentication fails if no or incorrect ACLs or rules are configured on the CAMS/IMC server even if EAD is enabled on the LNS.

The LNS can use different ACLs to filter packets from different iNode clients.

As a best practice, use EAD authentication for iNode clients on the Internet and use Portal authentication for iNode clients on a LAN.

Prerequisites

Make sure AAA, RADIUS, L2TP, Portal, and the security policy server are configured as required before you enable L2TP-based EAD.

For more information about AAA, RADIUS, and Portal, see *Security Configuration Guide*.

For more information about configuring the security policy server, see *IMC EAD Security Policy Manager Help*.

Procedure

1. Enter system view.

   `system-view`

2. Create a VT interface and enter its view

   `interface virtual-template` *`interface-number`*

3. Enable L2TP-based EAD.

ppp access-control enable

By default, L2TP-based EAD is disabled.

Display and maintenance commands for L2TP

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display L2TP tunnel information.	**display l2tp tunnel** [**statistics**]
Display L2TP session information.	**display l2tp session** [**statistics**]
Display information about temporary L2TP sessions.	**display l2tp session temporary**
Display information about virtual PPP interfaces.	**display interface** [**virtual-ppp** [*interface-number*]] [**brief** [**description** | **down**]]
Display VA pool information for L2TP.	**display l2tp va-pool** [**dynamic**]
Disconnect an L2TP tunnel.	**reset l2tp tunnel** { **id** *tunnel-id* | **name** *remote-name* }
Clear the statistics for virtual PPP interfaces.	**reset counters interface** [**virtual-ppp** [*interface-number*]]

L2TP configuration examples

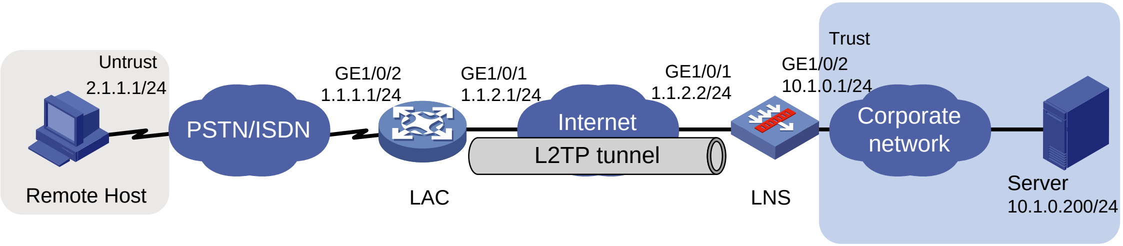
Example: Configuring a NAS-initiated L2TP tunnel

Network requirements

As shown in Figure 11, a PPP user is connected to an LNS through an LAC.

Set up an L2TP tunnel between the LAC and LNS to allow the PPP user to access the corporate network.

Figure 11 Network diagram



Configuration guidelines

You must add the Virtual-Template interface on the LNS to a security zone, and permit the traffic from security zone **Untrust** to security zone **Trust** and the traffic from security zone **Untrust** to security zone **Local**.

Configuration procedure

1. Configure the LAC:

# Create a local user named **vpdnuser**, set the password, and enable the PPP service.

```
<LAC> system-view
[LAC] local-user vpdnuser class network
[LAC-luser-network-vpdnuser] password simple Hello
[LAC-luser-network-vpdnuser] service-type ppp
[LAC-luser-network-vpdnuser] quit
```

# Configure local authentication for PPP users in ISP domain **system**.

```
[LAC] domain system
[LAC-isp-system] authentication ppp local
[LAC-isp-system] quit
```

# Configure CHAP authentication on interface GigabitEthernet 1/0/2.

```
[LAC] interface gigabitethernet 1/0/2
[LAC-GigabitEthernet1/0/2] ppp authentication-mode chap
[LAC-GigabitEthernet1/0/2] quit
```

# Enable L2TP.

```
[LAC] l2tp enable
```

# Create L2TP group 1 in LAC mode.

```
[LAC] l2tp-group 1 mode lac
```

# Configure the local tunnel name as **LAC**.

```
[LAC-l2tp1] tunnel name LAC
```

# Specify PPP user **vpdnuser** as the condition for the LAC to initiate tunneling requests.

```
[LAC-l2tp1] user fullusername vpdnuser
```

# Specify the LNS IP address as 1.1.2.2.

```
[LAC-l2tp1] lns-ip 1.1.2.2
```

# Enable tunnel authentication, and specify the tunnel authentication key as **aabbcc**.

```
[LAC-l2tp1] tunnel authentication
[LAC-l2tp1] tunnel password simple aabbcc
[LAC-l2tp1] quit
```

2. Configure the LNS:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<LNS> system-view
[LNS] interface gigabitethernet 1/0/1
[LNS-GigabitEthernet1/0/1] ip address 1.1.2.2 255.255.255.0
[LNS-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create a PPP address pool.

```
[LNS] ip pool aaa 192.168.0.10 192.168.0.20
[LNS] ip pool aaa gateway 192.168.0.1
```

# Create Virtual-Template 1, specify its PPP authentication mode as CHAP, assign an IP address to it, and use address pool **aaa** to assign IP addresses to the PPP users.

```
[LNS] interface virtual-template 1
[LNS-Virtual-Template1] ppp authentication-mode chap domain system
[LNS-Virtual-Template1] ip address 192.168.0.1 24
[LNS-Virtual-Template1] remote address pool aaa
[LNS-Virtual-Template1] quit
```

# Configure settings for routing. This example configures a static route, and the next hop in the route is 192.168.0.10, IP address that the LNS assigns to the LAC.

```
[LNS] ip route-static 2.1.1.1 24 192.168.0.10
```

# Add interfaces to security zones.

```
[LNS] security-zone name Untrust
[LNS-security-zone-Untrust] import interface Virtual-Template 1
[LNS-security-zone-Untrust] import interface gigabitethernet 1/0/1
[LNS-security-zone-Untrust] quit
[LNS] security-zone name Trust
[LNS-security-zone-Trust] import interface gigabitethernet 1/0/2
[LNS-security-zone-Trust] quit
```

# Configure a security policy rule named **l2tplocalout** to allow the LNS to send packets to the LAC.

```
[LNS] security-policy ip
[LNS-security-policy-ip] rule name l2tplocalout
[LNS-security-policy-ip-1-l2tplocalout] source-zone local
[LNS-security-policy-ip-1-l2tplocalout] destination-zone untrust
[LNS-security-policy-ip-1-l2tplocalout] source-ip-host 1.1.2.2
[LNS-security-policy-ip-1-l2tplocalout] destination-ip-host 1.1.2.1
[LNS-security-policy-ip-1-l2tplocalout] action pass
[LNS-security-policy-ip-1-l2tplocalout] quit
```

# Configure a security policy rule named **l2tplocalin** to allow the LNS to receive packets from the remote host.

```
[LNS-security-policy-ip] rule name l2tplocalin
[LNS-security-policy-ip-2-l2tplocalin] source-zone untrust
[LNS-security-policy-ip-2-l2tplocalin] destination-zone local
[LNS-security-policy-ip-2-l2tplocalin] source-ip-host 1.1.2.1
[LNS-security-policy-ip-2-l2tplocalin] destination-ip-host 1.1.2.2
[LNS-security-policy-ip-2-l2tplocalin] action pass
[LNS-security-policy-ip-2-l2tplocalin] quit
```

# Configure a security policy rule named **untrust-trust** to allow the LAC to access the server.

```
[LNS-security-policy-ip] rule name untrust-trust
[LNS-security-policy-ip-5-untrust-trust] source-zone untrust
[LNS-security-policy-ip-5-untrust-trust] destination-zone trust
[LNS-security-policy-ip-5-untrust-trust] source-ip-range 192.168.0.10 192.168.0.20
[LNS-security-policy-ip-5-untrust-trust] destination-ip-host 10.1.0.200
[LNS-security-policy-ip-5-untrust-trust] action pass
[LNS-security-policy-ip-5-untrust-trust] quit
```

# Configure a security policy rule named **trust-untrust** to allow the server to send packets to the LAC.

```
[LNS-security-policy-ip] rule name trust-untrust
[LNS-security-policy-ip-6-trust-untrust] source-zone trust
[LNS-security-policy-ip-6-trust-untrust] destination-zone untrust
[LNS-security-policy-ip-6-trust-untrust] source-ip-host 10.1.0.200
[LNS-security-policy-ip-6-trust-untrust] destination-ip-range 192.168.0.10
192.168.0.20
[LNS-security-policy-ip-6-trust-untrust] action pass
[LNS-security-policy-ip-6-trust-untrust] quit
[LNS-security-policy-ip] quit
```

# Create a local user named **vpdnuser**, set the password, and enable the PPP service.

```
[LNS] local-user vpdnuser class network
[LNS-luser-network-vpdnuser] password simple Hello
[LNS-luser-network-vpdnuser] service-type ppp
[LNS-luser-network-vpdnuser] quit
```

# Configure local authentication for PPP users in ISP domain **system**.

```
[LNS] domain system
[LNS-isp-system] authentication ppp local
[LNS-isp-system] quit
```

# Enable L2TP.

```
[LNS] l2tp enable
```

# Create L2TP group 1 in LNS mode.

```
[LNS] l2tp-group 1 mode lns
```

# Configure the local tunnel name as **LNS**.

```
[LNS-l2tp1] tunnel name LNS
```

# Specify Virtual-Template 1 for receiving calls from an LAC.

```
[LNS-l2tp1] allow l2tp virtual-template 1 remote LAC
```

# Enable tunnel authentication, and specify the tunnel authentication key as **aabbcc**.

```
[LNS-l2tp1] tunnel authentication
[LNS-l2tp1] tunnel password simple aabbcc
[LNS-l2tp1] quit
```

3. On the remote system, enter **vpdnuser** as the username and **Hello** as the password in the dial-up network window to dial a connection.

Verifying the configuration

After the dial-up connection is established, the remote system can obtain an IP address and can ping the private IP address of the LNS.

# On the LNS, use the **display l2tp tunnel** command to check the established L2TP tunnels.

```
[LNS] display l2tp tunnel
LocalTID RemoteTID State         Sessions RemoteAddress    RemotePort RemoteName
196      3542      Established   1        1.1.2.1          1701       LAC
```

# On the LNS, use the **display l2tp session** command to check the established L2TP sessions.

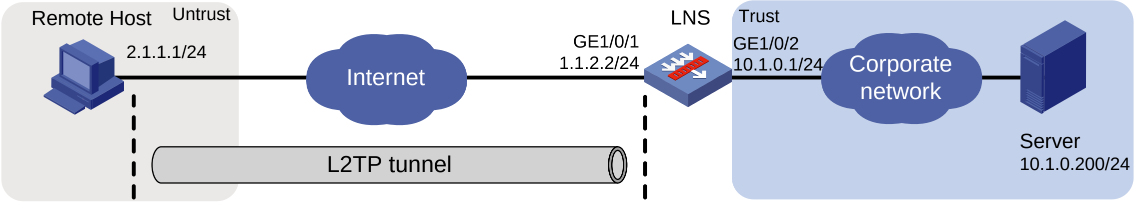
```
[LNS] display l2tp session
LocalSID      RemoteSID      LocalTID      State
2041          64             196           Established
```

Example: Configuring a client-initiated L2TP tunnel

Network requirements

As shown in Figure 12, a PPP user directly initiates a tunneling request to the LNS to access the corporate network.

Figure 12 Network diagram



Configuration guidelines

You must add the Virtual-Template interface on the LNS to a security zone, and permit the traffic from security zone **Untrust** to security zone **Trust** and the traffic from security zone **Untrust** to security zone **Local**.

Configuration procedure

1. Configure the LNS:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<LNS> system-view
[LNS] interface gigabitethernet 1/0/1
[LNS-GigabitEthernet1/0/1] ip address 1.1.2.2 255.255.255.0
[LNS-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create a PPP address pool.

```
[LNS] ip pool aaa 192.168.0.10 192.168.0.20
[LNS] ip pool aaa gateway 192.168.0.1
```

# Create Virtual-Template 1, specify its PPP authentication mode as CHAP, assign an IP address to it, and use address pool **aaa** to assign IP addresses to the PPP users.

```
[LNS] interface virtual-template 1
[LNS-Virtual-Template1] ppp authentication-mode chap domain system
[LNS-Virtual-Template1] ip address 192.168.0.1 24
[LNS-Virtual-Template1] remote address pool aaa
[LNS-Virtual-Template1] quit
```

# Configure settings for routing. This example configures a static route, and the next hop in the route is 1.1.2.3.

```
[LNS] ip route-static 2.1.1.1 24 1.1.2.3
```

# Add interfaces to security zones.

```
[LNS] security-zone name Untrust
[LNS-security-zone-Untrust] import interface Virtual-Template 1
[LNS-security-zone-Untrust] import interface gigabitethernet 1/0/1
[LNS-security-zone-Untrust] quit
[LNS] security-zone name Trust
[LNS-security-zone-Trust] import interface gigabitethernet 1/0/2
[LNS-security-zone-Trust] quit
```

# Configure a security policy rule named **l2tplocalout** to allow the LNS to send packets to the remote host.

```
[LNS] security-policy ip
[LNS-security-policy-ip] rule name l2tplocalout
[LNS-security-policy-ip-1-l2tplocalout] source-zone local
[LNS-security-policy-ip-1-l2tplocalout] destination-zone untrust
[LNS-security-policy-ip-1-l2tplocalout] source-ip-host 1.1.2.2
```

```
[LNS-security-policy-ip-1-l2tplocalout] destination-ip-host 2.1.1.1
[LNS-security-policy-ip-1-l2tplocalout] action pass
[LNS-security-policy-ip-1-l2tplocalout] quit
```

# Configure a security policy rule named **l2tplocalin** to allow the LNS to receive packets from the remote host.

```
[LNS-security-policy-ip] rule name l2tplocalin
[LNS-security-policy-ip-2-l2tplocalin] source-zone untrust
[LNS-security-policy-ip-2-l2tplocalin] destination-zone local
[LNS-security-policy-ip-2-l2tplocalin] source-ip-host 2.1.1.1
[LNS-security-policy-ip-2-l2tplocalin] destination-ip-host 1.1.2.2
[LNS-security-policy-ip-2-l2tplocalin] action pass
[LNS-security-policy-ip-2-l2tplocalin] quit
```

# Configure a security policy rule named **untrust-trust** to allow the remote host to access the server.

```
[LNS-security-policy-ip] rule name untrust-trust
[LNS-security-policy-ip-5-untrust-trust] source-zone untrust
[LNS-security-policy-ip-5-untrust-trust] destination-zone trust
[LNS-security-policy-ip-5-untrust-trust] source-ip-range 192.168.0.10 192.168.0.20
[LNS-security-policy-ip-5-untrust-trust] destination-ip-host 10.1.0.200
[LNS-security-policy-ip-5-untrust-trust] action pass
[LNS-security-policy-ip-5-untrust-trust] quit
```

# Configure a security policy rule named **trust-untrust** to allow the server to send packets to the remote host.

```
[LNS-security-policy-ip] rule name trust-untrust
[LNS-security-policy-ip-6-trust-untrust] source-zone trust
[LNS-security-policy-ip-6-trust-untrust] destination-zone untrust
[LNS-security-policy-ip-6-trust-untrust] source-ip-host 10.1.0.200
[LNS-security-policy-ip-6-trust-untrust] destination-ip-range 192.168.0.10
192.168.0.20
[LNS-security-policy-ip-6-trust-untrust] action pass
[LNS-security-policy-ip-6-trust-untrust] quit
[LNS-security-policy-ip] quit
```

# Create a local user named **vpdnuser**, set the password, and enable the PPP service.

```
[LNS] local-user vpdnuser class network
[LNS-luser-network-vpdnuser] password simple Hello
[LNS-luser-network-vpdnuser] service-type ppp
[LNS-luser-network-vpdnuser] quit
```

# Configure local authentication for PPP users in ISP domain **system**.

```
[LNS] domain system
[LNS-isp-system] authentication ppp local
[LNS-isp-system] quit
```

# Enable L2TP.

```
[LNS] l2tp enable
```

# Create L2TP group 1 in LNS mode.

```
[LNS] l2tp-group 1 mode lns
```

# Configure the local tunnel name as **LNS**.

```
[LNS-l2tp1] tunnel name LNS
```

# Specify Virtual-Template 1 for receiving calls.

```
[LNS-l2tp1] allow l2tp virtual-template 1
```

# Disable tunnel authentication.

```
[LNS-l2tp1] undo tunnel authentication
[LNS-l2tp1] quit
```

2. Configure the remote host:

   # Configure the IP address of the remote host as 2.1.1.1, and configure a route to the LNS (1.1.2.2).

   # Create a virtual private network connection by using the Windows system, or install the L2TP LAC client software.

   # Complete the following configuration procedure (the procedure depends on the client software):

   - Specify the PPP username as **vpdnuser** and the password as **Hello**.
   - Specify the Internet interface address of the security gateway as the IP address of the LNS. In this example, the Ethernet interface for the tunnel on the LNS has an IP address of 1.1.2.2.
   - Modify the connection attributes: set the protocol to **L2TP**, the encryption attribute to **customized**, and the authentication mode to **CHAP**.

Verifying the configuration

# On the remote host, initiate the L2TP connection. After the connection is established, the remote host can obtain the IP address 192.168.0.2 and ping the private IP address of the LNS (192.168.0.1).

# On the LNS, use the **`display l2tp session`** command to check the established L2TP session.

```
[LNS] display l2tp session
LocalSID    RemoteSID    LocalTID    State
39945       1            37263       Established
```

# On the LNS, use the **`display l2tp tunnel`** command to check the established L2TP tunnel.

```
[LNS] display l2tp tunnel
LocalTID RemoteTID State         Sessions RemoteAddress    RemotePort RemoteName
37263    4         Established   1        2.1.1.1          1701       PC
```

# On the remote host, verify that you can ping the server 10.1.0.200 in the corporate network.

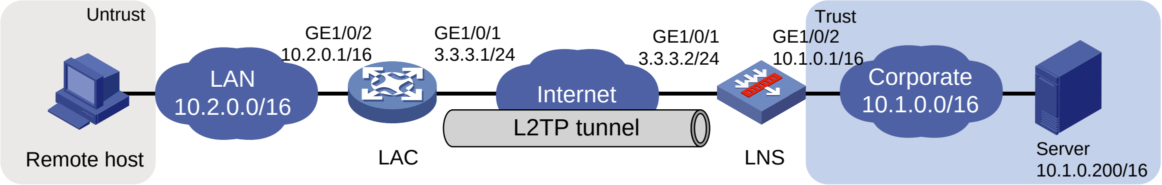
```
C:\> ping 10.1.0.200

Pinging 10.1.0.200 with 32 bytes of data:
Reply from 10.1.0.200: bytes=32 time<1ms TTL=254
Reply from 10.1.0.200: bytes=32 time<1ms TTL=254
Reply from 10.1.0.200: bytes=32 time<1ms TTL=254
Reply from 10.1.0.200: bytes=32 time<1ms TTL=254
Ping statistics for 10.1.0.200:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Example: Configuring a LAC-auto-initiated L2TP tunnel

Network requirements

As shown in Figure 13, configure the LAC to establish an L2TP tunnel with the LNS in LAC-auto-initiated mode. When the PPP user initiates a connection, it uses the established tunnel to access the corporate network.

Figure 13 Network diagram



Configuration guidelines

You must add the Virtual-Template interface on the LNS to a security zone, and permit the traffic from security zone **Untrust** to security zone **Trust** and the traffic from security zone **Untrust** to security zone **Local**.

Configuration procedure

1. Configure the LAC:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<LAC> system-view
[LAC] interface gigabitethernet 1/0/1
[LAC-GigabitEthernet1/0/1] ip address 3.3.3.1 255.255.0.0
[LAC-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create Virtual-PPP 1. Configure its username and password as **vpdnuser** and **Hello** and PPP authentication as PAP.

```
[LAC] interface virtual-ppp 1
[LAC-Virtual-PPP1] ip address ppp-negotiate
[LAC-Virtual-PPP1] ppp pap local-user vpdnuser password simple Hello
[LAC-Virtual-PPP1] quit
```

# Configure a static route so that packets destined for the corporate network will be forwarded through the L2TP tunnel.

```
[LAC] ip route-static 10.1.0.0 16 virtual-ppp 1
```

# Enable L2TP.

```
[LAC] l2tp enable
```

# Create L2TP group 1 in LAC mode.

```
[LAC] l2tp-group 1 mode lac
```

# Configure the local tunnel name as **LAC**, and specify the IP address of the tunnel peer (LNS) as 3.3.3.2.

```
[LAC-l2tp1] tunnel name LAC
[LAC-l2tp1] lns-ip 3.3.3.2
```

# Enable tunnel authentication, and configure the authentication key as **aabbcc**.

```
[LAC-l2tp1] tunnel authentication
[LAC-l2tp1] tunnel password simple aabbcc
[LAC-l2tp1] quit
```

# Trigger the LAC to establish an L2TP tunnel with the LNS.

```
[LAC] interface virtual-ppp 1
[LAC-Virtual-PPP1] l2tp-auto-client l2tp-group 1
[LAC-Virtual-PPP1] quit
```

2. Configure the LNS:

# Assign an IP address to GigabitEthernet 1/0/1.

```
<LNS> system-view
```

```
[LNS] interface gigabitethernet 1/0/1
[LNS-GigabitEthernet1/0/1] ip address 3.3.3.2 255.255.0.0
[LNS-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

# Create Virtual-Template 1, assign an IP address to it, specify its PPP authentication mode as PAP, and assign the IP address 192.168.0.10 to the PPP user.

```
 [LNS] interface virtual-template 1
[LNS-Virtual-Template1] ip address 192.168.0.1 24
[LNS-Virtual-Template1] ppp authentication-mode pap
[LNS-Virtual-Template1] remote address 192.168.0.10
[LNS-Virtual-Template1] quit
```

# Configure settings for routing. This example configures a static route, and the next hop in the routes is 192.168.0.10.

```
[LNS] ip route-static 10.2.0.0 16 192.168.0.10
```

# Add interfaces to security zones.

```
[LNS] security-zone name Untrust
[LNS-security-zone-Untrust] import interface Virtual-Template 1
[LNS-security-zone-Untrust] import interface gigabitethernet 1/0/1
[LNS-security-zone-Untrust] quit
[LNS] security-zone name Trust
[LNS-security-zone-Trust] import interface gigabitethernet 1/0/2
[LNS-security-zone-Trust] quit
```

# Configure a security policy rule named **l2tplocalout** to allow the LNS to send packets to the LAC.

```
[LNS] security-policy ip
[LNS-security-policy-ip] rule name l2tplocalout
[LNS-security-policy-ip-1-l2tplocalout] source-zone local
[LNS-security-policy-ip-1-l2tplocalout] destination-zone untrust
[LNS-security-policy-ip-1-l2tplocalout] source-ip-host 3.3.3.2
[LNS-security-policy-ip-1-l2tplocalout] destination-ip-host 3.3.3.1
[LNS-security-policy-ip-1-l2tplocalout] action pass
[LNS-security-policy-ip-1-l2tplocalout] quit
```

# Configure a security policy rule named **l2tplocalin** to allow the LNS to receive packets from the LAC.

```
[LNS-security-policy-ip] rule name l2tplocalin
[LNS-security-policy-ip-2-l2tplocalin] source-zone untrust
[LNS-security-policy-ip-2-l2tplocalin] destination-zone local
[LNS-security-policy-ip-2-l2tplocalin] source-ip-host 3.3.3.1
[LNS-security-policy-ip-2-l2tplocalin] destination-ip-host 3.3.3.2
[LNS-security-policy-ip-2-l2tplocalin] action pass
[LNS-security-policy-ip-2-l2tplocalin] quit
```

# Configure a security policy rule named **untrust-trust** to allow the LAC to access the server.

```
[LNS-security-policy-ip] rule name untrust-trust
[LNS-security-policy-ip-5-untrust-trust] source-zone untrust
[LNS-security-policy-ip-5-untrust-trust] destination-zone trust
[LNS-security-policy-ip-5-untrust-trust] source-ip-host 192.168.0.10
[LNS-security-policy-ip-5-untrust-trust] destination-ip-host 10.1.0.200
[LNS-security-policy-ip-5-untrust-trust] action pass
```

```
[LNS-security-policy-ip-5-untrust-trust] quit
```

# Configure a security policy rule named **trust-untrust** to allow the server to send packets to the LAC.

```
[LNS-security-policy-ip] rule name trust-untrust
[LNS-security-policy-ip-6-trust-untrust] source-zone trust
[LNS-security-policy-ip-6-trust-untrust] destination-zone untrust
[LNS-security-policy-ip-6-trust-untrust] source-ip-host 10.1.0.200
[LNS-security-policy-ip-6-trust-untrust] destination-ip-host 192.168.0.10
[LNS-security-policy-ip-6-trust-untrust] action pass
[LNS-security-policy-ip-6-trust-untrust] quit
[LNS-security-policy-ip] quit
```

# Create a local user named **vpdnuser**, set the password, and enable the PPP service.

```
[LNS] local-user vpdnuser class network
[LNS-luser-network-vpdnuser] password simple Hello
[LNS-luser-network-vpdnuser] service-type ppp
[LNS-luser-network-vpdnuser] quit
```

# Configure local authentication for PPP users in ISP domain **system**.

```
[LNS] domain system
[LNS-isp-system] authentication ppp local
[LNS-isp-system] quit
```

# Enable L2TP, and create L2TP group 1 in LNS mode.

```
[LNS] l2tp enable
[LNS] l2tp-group 1 mode lns
```

# Configure the local tunnel name as **LNS**, and specify Virtual-Template 1 for receiving tunneling requests from an LAC.

```
[LNS-l2tp1] tunnel name LNS
[LNS-l2tp1] allow l2tp virtual-template 1 remote LAC
```

# Enable tunnel authentication, and configure the authentication key as **aabbcc**.

```
[LNS-l2tp1] tunnel authentication
[LNS-l2tp1] tunnel password simple aabbcc
[LNS-l2tp1] quit
```

3. On the remote host, configure the LAC as the gateway.

Verifying the configuration

# On the LNS, use the **`display l2tp session`** command to display the established L2TP session.

```
[LNS] display l2tp session
LocalSID    RemoteSID    LocalTID    State
21073       11183        52525       Established
```

# On the LNS, use the **`display l2tp tunnel`** command to display the established L2TP tunnel.

```
[LNS] display l2tp tunnel
LocalTID RemoteTID State        Sessions RemoteAddress   RemotePort RemoteName
52525    33375     Established  1        3.3.3.1         1701       LAC
```

# On the remote host, verify that you can ping the server 10.1.0.200 in the corporate network.

```
C:\> ping 10.1.0.200

Pinging 10.1.0.200 with 32 bytes of data:
Reply from 10.1.0.200: bytes=32 time<1ms TTL=253
Reply from 10.1.0.200: bytes=32 time<1ms TTL=253
Reply from 10.1.0.200: bytes=32 time<1ms TTL=253
Reply from 10.1.0.200: bytes=32 time<1ms TTL=253
Ping statistics for 10.1.0.200:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Troubleshooting L2TP

Failure to access the private network

Symptom

The remote system cannot access the private network.

Solution

To resolve the problem:

1. Verify the following items to avoid tunnel setup failures:
   - The address of the LNS is configured correctly on the LAC. For more information, see the **lns-ip** command.
   - The LNS can accept L2TP tunneling requests from the LAC. For more information, see the **allow** command.
   - Tunnel authentication is enabled on both the LAC and the LNS, and the tunnel authentication keys configured on the two sides match.
2. Verify the following items to avoid PPP negotiation failures:
   - Usernames and passwords are correctly configured on the LAC and LNS.
   - IP address negotiation settings are correct on the remote system and LNS.
   - The authentication type is consistent. For example, the default authentication type for a VPN connection created on Windows 2000 is MS-CHAP. If the peer does not support MS-CHAP, change the authentication type to CHAP on Windows 2000.

Data transmission failure

Symptom

Data transmission fails. A connection is established, but data cannot be transmitted. For example, the LAC and LNS cannot ping each other.

Solution

To resolve the problem:

1. Use the **display ip routing-table** command on the LAC and LNS to verify that the LAC has a route to the private network behind the LNS, and vice versa. If no route is available, configure a static route or a dynamic routing protocol.
2. Increase the link bandwidth to enhance the link availability.

   Internet backbone congestion and high packet loss ratio might cause data transmission failures. L2TP data transmission is based on UDP, which does not provide the packet error control feature. If the line is unstable, the LAC and LNS might be unable to ping each other.

Contents

Configuring ADVPN

About ADVPN

Auto Discovery Virtual Private Network (ADVPN) enables enterprise branches that use dynamic public addresses to establish a VPN network. ADVPN uses the VPN Address Management (VAM) protocol to collect, maintain, and distribute dynamic public addresses.

VAM protocol

VAM uses the client/server model. All VAM clients register their public addresses with the VAM server. A VAM client obtains the public addresses of other clients from the server to establish ADVPN tunnels.

ADVPN structures

ADVPN uses domains to identify VPNs. VAM clients in a VPN must be assigned to the same ADVPN domain. A VAM client can belong to only one ADVPN domain. A VAM server can serve multiple ADVPN domains and manage their clients.

VAM clients include hubs and spokes.

- **Hub**—A hub is the exchange center of routing information. A hub in a hub-spoke network is also a data forwarding center.
- **Spoke**—A spoke is the gateway of a branch. It does not forward data received from other ADVPN nodes.

ADVPN supports full-mesh, hub-spoke, and hub-group structures.

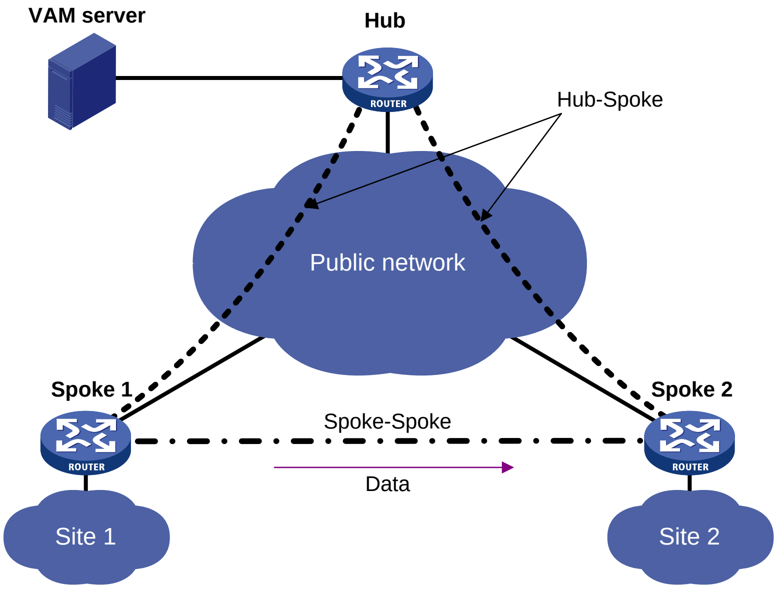
Full-mesh ADVPN

In a full-mesh ADVPN, spokes can directly communicate with each other. The hub acts as the route exchange center.

As shown in Figure 1, the spokes register with the VAM server and get hub information in the ADVPN domain. Then, they establish permanent tunnels to the hub.

Any two spokes can establish a dynamic tunnel to directly exchange data. The tunnel is deleted if no data exists during the idle timeout time.

Figure 1 Full-mesh ADVPN

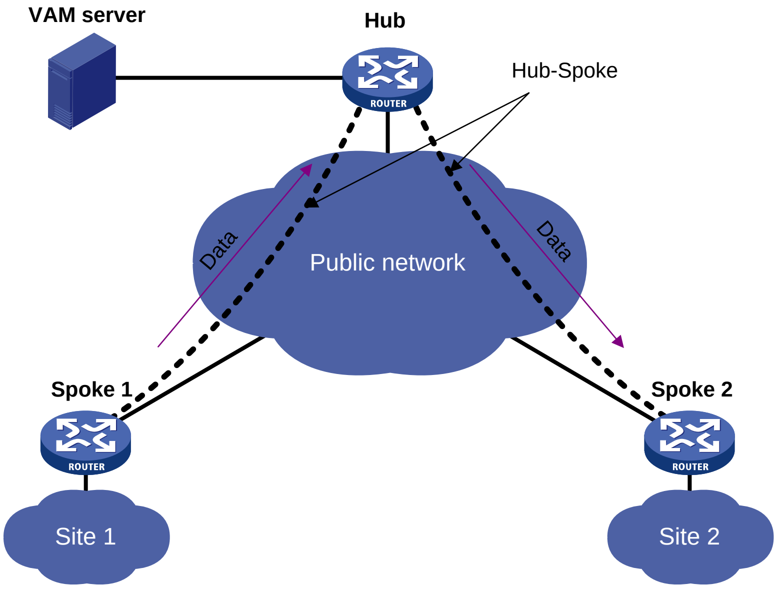


Hub-spoke ADVPN

In a hub-spoke ADVPN, spokes communicate with each other through the hub. The hub acts as both the route exchange center and data forwarding center.

As shown in Figure 2, each spoke establishes a permanent tunnel to the hub. Spokes communicate with each other through the hub.

Figure 2 Hub-spoke ADVPN



Hub-group ADVPN

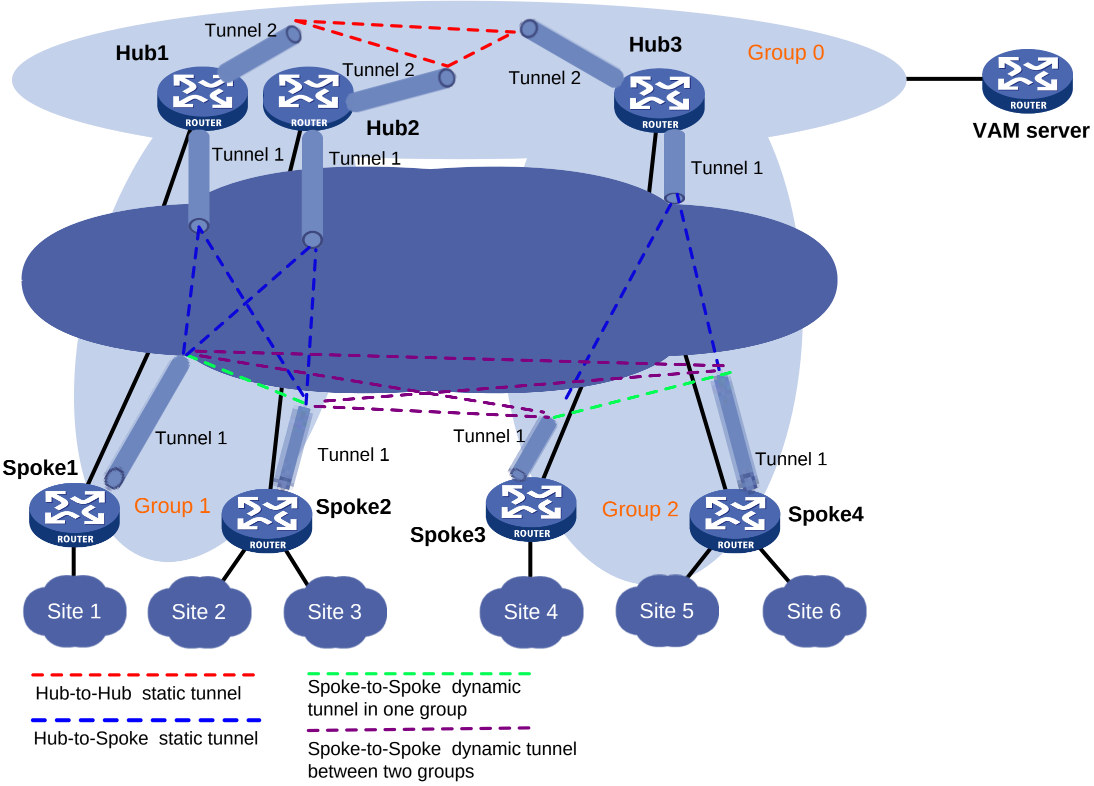
A hub-group ADVPN can accommodate more ADVPN clients. This allows one hub to manage all clients. As shown in Figure 3, a hub-group ADVPN contains multiple hub groups. Each hub group has one or multiple hubs and spokes.

Follow these guidelines to classify hub groups:

- All hubs must belong to the backbone hub group. This hub group forms the full-mesh backbone area. All hubs obtain information about other hubs from the VAM server and establish permanent ADVPN tunnels to each other.
- Spokes must belong to non-backbone hub groups. Each non-backbone hub group includes at least one hub and uses either the full-mesh or hub-spoke structure. Spokes obtain hub information in the ADVPN domain from the VAM server, and establish permanent tunnels to the hub. Spokes can establish tunnels only to the hubs in the hub group.

Tunnel establishment and data forwarding in a hub group depend on the network structure. Inter-group communications between spokes need to pass the hubs of the groups. To reduce the pressure on hubs during inter-group communications, you can allow spokes in different hub groups to establish a dynamic tunnel. The dynamic tunnel is deleted if no data exists during the idle timeout time.

Figure 3 Hub-group ADVPN



How ADVPN operates

The VAM server must have a static public address. VAM clients have both a public address and a private address. The public address is the address of the interface connected to the public network. It can be manually configured or dynamically assigned. The private address is the address of the ADVPN tunnel interface. It must be manually configured. All the private addresses of clients in an ADVPN domain must belong to the same network segment.

ADVPN includes the following phases:

- Connection initialization.
- Registration.
- Tunnel establishment.
- Route learning and packet forwarding.

Connection initialization

As shown in Figure 4, a client and a server perform the following operations to initialize a connection:

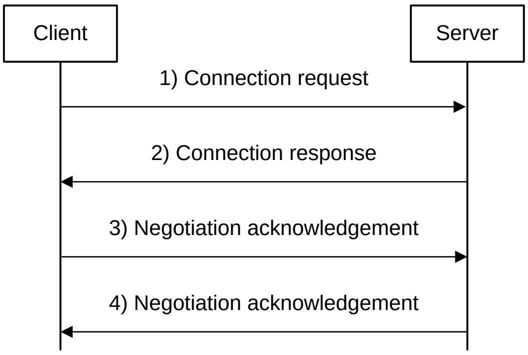
1. The client sends encryption and authentication algorithms to the server in a connection request.
2. The server compares its algorithms in descending order of priority with the algorithms sent by the client.
3. The server sends the matching algorithms to the client.

   If no match is found, the negotiation fails.
4. The server and the client generate encryption and authentication keys based on the preshared key.

   If authentication and encryption are not needed, they do not generate keys.
5. The server and the client exchange negotiation acknowledgment packets protected by using the keys.
6. The server and the client use the keys to protect subsequent packets if they can restore the protected negotiation acknowledgment packets.

   If they cannot restore the packets, the negotiation fails.

Figure 4 Connection initialization process


Client
Server
1) Connection request
2) Connection response
3) Negotiation acknowledgement
4) Negotiation acknowledgement

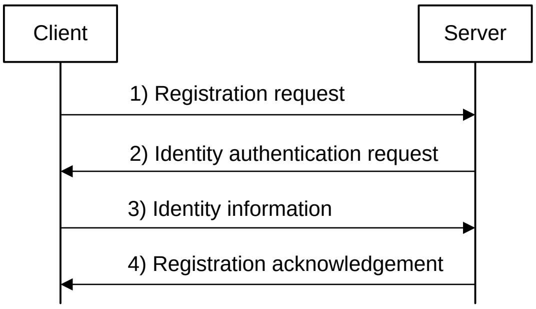
Registration

Figure 5 shows the following registration process:

1. The client sends the server a registration request that includes its public address, private address, and the connected private network.
2. The server sends the client an identity authentication request that specifies the authentication algorithm.

   If authentication is not required, the server directly registers the client and sends the client a registration acknowledgement. VAM supports both PAP and CHAP authentication.
3. The client submits its identity information to the server.
4. The server performs authentication and accounting for the client through the AAA server.
5. The server sends the client a registration acknowledgement that includes hub information.

Figure 5 Registration process


Client
Server
1) Registration request
2) Identity authentication request
3) Identity information
4) Registration acknowledgement

Tunnel establishment

A spoke can establish permanent tunnels to any number of hubs. Hubs in an ADVPN domain must establish permanent tunnels.

Figure 6 shows the tunnel establishment process.

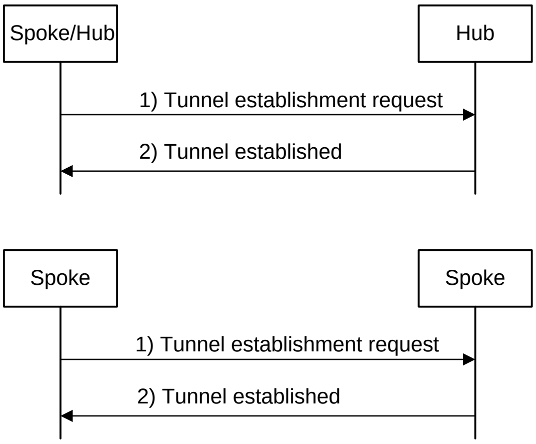
1. The initiator originates a tunnel establishment request.
   - **To establish a hub-spoke tunnel:**

     The spoke checks whether a tunnel to each hub exists. If not, the spoke sends a tunnel establishment request to the hub.
   - **To establish a hub-hub tunnel:**

     The hub checks whether a tunnel to each peer hub exists. If not, the hub sends a tunnel establishment request to the peer hub.
   - **To establish a spoke-spoke tunnel:**

     In a full-mesh network, when a spoke receives a data packet but finds no tunnel for forwarding the packet, it sends an address resolution request to the server. After receiving the resolved address, the spoke sends a tunnel establishment request to the peer spoke.
2. The receiver saves tunnel information in the request and sends a response to the sender.

Figure 6 Tunnel establishment process



Route learning and packet forwarding

ADVPN nodes use the following methods to learn private routes:

- **Static or dynamic routing**—It must be configured for private networks and ADVPN tunnel interfaces to ensure connectivity among private networks. A dynamic routing protocol discovers neighbors, updates routes, and establishes a routing table over ADVPN tunnels. From the perspective of private networks, ADVPN tunnels are links that connect different private networks. The routing protocol exchanges routes between hub and hub, and between hub and spoke. It does not directly exchange routes between spoke and spoke.

When a spoke receives a packet destined to a remote private network, it performs the following operations to forward the packet:

- **a.** Locates the private next hop from the routing table.
- **b.** Uses the private next hop to obtain the corresponding public address from the VAM server.
- **c.** Sends the packet to the public address over the ADVPN tunnel.

Full-mesh and hub-spoke structures are determined by routing. If the next hop is a spoke, the structure is full-mesh. If the next hop is a hub, the structure is hub-spoke.

- **Registration and query from the VAM server**—VAM clients register information about the connected private networks on the VAM server.

  When a spoke receives a packet destined to a remote private network, it performs the following operations to forward the packet:

  - **a.** Sends the destination address of the packet to the VAM server.
  - **b.** Queries the VAM server for information about the ADVPN node (public and private addresses of the node) connected to the remote private network.
  - **c.** Generates a route to the remote private network through the ADVPN node.
  - **d.** Sends the packet to the public address of the ADVPN node over the ADVPN tunnel.

If both methods are used, the spoke sends both the private next hop and the destination address of the packet to the VAM server. The VAM server preferentially obtains the private network according to the destination address. If the route to the remote private network is learned by using both methods, the route with a lower preference is used.

NAT traversal

An ADVPN tunnel can traverse a NAT gateway.

- If only the tunnel initiator resides behind a NAT gateway, a spoke-spoke tunnel can be established through the NAT gateway.
- If the tunnel receiver resides behind a NAT gateway, packets must be forwarded by a hub before the receiver originates a tunnel establishment request. If the NAT gateway uses Endpoint-Independent Mapping, a spoke-spoke tunnel can be established through the NAT gateway.
- If both ends reside behind a NAT gateway, no tunnel can be established and packets between them must be forwarded by a hub.

ADVPN tasks at a glance

Configure ADVPN in the order of VAM servers, hubs, and spokes. Whether an ADVPN node is a hub or a spoke is specified on the VAM servers.

To configure ADVPN, perform the following tasks:

1. Configuring the VAM server
2. Configuring ADVPN nodes
   - **a.** Configuring the VAM client
   - **b.** Configuring routing
   - **c.** Configuring an ADVPN tunnel interface
   - **d.** (Optional.) Configuring IPsec for ADVPN tunnels
   - **e.** (Optional.) Enabling ADVPN logging

Configuring the VAM server

Hardware compatibility with VAM server

Models	VAM server compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080	Yes
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

VAM server tasks at a glance

To configure the VAM server, perform the following tasks:

1. Creating an ADVPN domain
2. Enabling the VAM server
3. Configuring a preshared key for the VAM server
4. Configuring hub groups
5. (Optional.) Setting the port number of the VAM server
6. (Optional.) Specifying authentication and encryption algorithms for the VAM server
7. (Optional.) Configuring an authentication method
8. (Optional.) Configuring keepalive parameters
9. (Optional.) Setting the retry timer

Creating an ADVPN domain

1. Enter system view.
   system-view
2. Create an ADVPN domain and enter ADVPN domain view.
   vam server advpn-domain *domain-name* **id** *domain-id*

Enabling the VAM server

1. Enter system view.
   system-view
2. Enable the VAM server. Choose one of the following tasks:
   - In system view, enable the VAM server for one or all ADVPN domains.
     vam server enable [**advpn-domain** *domain-name*]
   - Execute the following commands in sequence to enable the VAM server for an ADVPN domain:
     vam server advpn-domain *domain-name* [**id** *domain-id*]
     server enable

   By default, the VAM server is disabled.

Configuring a preshared key for the VAM server

About this task

The preshared key is used to generate initial encryption and authentication keys during connection initialization. It is also used to generate encryption and authentication keys for subsequent packets if encryption and authentication are needed.

If the preshared key on the VAM server is different than the VAM clients, packet decryption and integrity verification will fail. The VAM server and clients cannot establish connections.

Restrictions and guidelines

The VAM server and the VAM clients in the same ADVPN domain must have the same preshared key.

Procedure

1. Enter system view.

   `system-view`

2. Enter ADVPN domain view.

   `vam server advpn-domain` *`domain-name`* [**`id`** *`domain-id`*]

3. Configure a preshared key for the VAM server.

   `pre-shared-key` { **`cipher`** | **`simple`** } *`string`*

   By default, no preshared key is configured.

Configuring hub groups

About this task

Hub groups apply to large ADVPN networks. You can classify spokes to different hub groups, and specify one or more hubs for each group. The VAM server assigns a client only the hub information of the client's own hub group. A client establishes permanent ADVPN tunnels only to the hubs in its own hub group.

By default, spokes are not allowed to establish direct tunnels. If an ACL is specified to control establishing spoke-spoke tunnels, the VAM server assigns the specified ACL to an online hub. The hub uses the ACL to match received packets. If a match is found, the hub sends a redirect packet to the spoke that sent the packet. Then, the spoke sends the VAM server the destination address of the packet, obtains the remote spoke information, and establishes a direct tunnel to the remote spoke.

After a spoke-spoke tunnel is established, the spokes directly exchange packets.

When a VAM client registers with the VAM server, the VAM server selects a hub group for the client as follows:

1. The server matches the private address of the client against the private addresses of hubs in different hub groups in lexicographic order.
2. If a match is found, the server assigns the client to the hub group as a hub.
3. If no match is found, the server matches the client's private address against the private addresses of spokes in different hub groups in lexicographic order.
4. If a match is found, the server assigns the client to the hub group as a spoke.
5. If no match is found, the registration fails.

Procedure

1. Enter system view.

   `system-view`

2. Enter ADVPN domain view.

vam server advpn-domain *domain-name* [**id** *domain-id*]

3. Create a hub group and enter hub group view.

   hub-group *group-name*

4. Configure a hub private address.

   IPv4:

   hub private-address *private-ip-address* [**public-address** { *public-ipv4-address* | *public-ipv6-address* } [**advpn-port** *port-number*]]

   IPv6:

   hub ipv6 private-address *private-ipv6-address* [**public-address** { *public-ipv4-address* | *public-ipv6-address* } [**advpn-port** *port-number*]]

   By default, no hub private addresses are configured.

   A hub group must have a minimum of one hub private address.

5. Configure a spoke private address range.

   IPv4:

   spoke private-address { **network** *ip-address* { *mask-length* | *mask* } | **range** *start-ipv4-address end-ipv4-address* }

   IPv6:

   spoke ipv6 private-address { **network** *prefix prefix-length* | **range** *start-ipv6-address end-ipv6-address* }

   By default, no spoke private address ranges are configured.

   You can configure multiple spoke private IPv4 and IPv6 address ranges in a hub group.

6. Specify an ACL to control establishing spoke-to-spoke tunnels.

   IPv4:

   shortcut interest { **acl** { *acl-number* | **name** *acl-name* } **all** }

   IPv6:

   shortcut ipv6 interest { **acl** { *ipv6-acl-number* | **name** *ipv6-acl-name* } | **all** }

   By default, spokes are not allowed to establish direct tunnels.

Setting the port number of the VAM server

Restrictions and guidelines

The port number of the VAM server must be the same as that configured on the VAM clients.

Procedure

1. Enter system view.

   system-view

2. Set the port number of the VAM server.

   vam server listen-port *port-number*

   The default port number is 18000.

Specifying authentication and encryption algorithms for the VAM server

About this task

The VAM server uses the specified algorithms to negotiate with the VAM client.

The VAM server and client use SHA-1 and AES-CBC-128 during connection initialization, and use the negotiated algorithms after connection initialization.

Restrictions and guidelines

The algorithm specified earlier in a command line has a higher priority.

The configuration of the commands that specify authentication and encryption algorithms does not affect registered VAM clients. It applies to subsequently registered VAM clients.

Procedure

1. Enter system view.

   system-view

2. Enter ADVPN domain view.

   vam server advpn-domain *domain-name* [**id** *domain-id*]

3. Specify authentication algorithms.

   authentication-algorithm { **aes-xcbc-mac** | **md5** | **none** | **sha-1** | **sha-256** } *

   The default authentication algorithm is SHA-1.

4. Specify encryption algorithms.

   encryption-algorithm { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256** | **des-cbc** | **none** } *

   The default encryption algorithms are AES-CBC-256, AES-CBC-192, AES-CBC-128, AES-CTR-256, AES-CTR-192, AES-CTR-128, 3DES-CBC, and DES-CBC in descending order of priority.

Configuring an authentication method

About this task

The VAM server uses the specified method to authenticate clients in the ADVPN domain. The authentication method includes none authentication and AAA authentication. If AAA is used, the VAM server supports PAP and CHAP authentication. Only VAM clients that pass identity authentication can access the ADVPN domain. For information about AAA configuration on the VAM server, see *Security Configuration Guide*.

Restrictions and guidelines

If the specified ISP domain does not exist, the authentication will fail.

A newly configured authentication method does not affect registered VAM clients. It applies to subsequently registered VAM clients.

Procedure

1. Enter system view.

   system-view

2. Enter ADVPN domain view.

   vam server advpn-domain *domain-name* [**id** *domain-id*]

3. Specify an authentication method.

   authentication-method { **none** | { **chap** | **pap** } [**domain** *isp-name*] }

   By default, the authentication method is CHAP, and the default ISP domain is used.

Configuring keepalive parameters

About this task

Keepalive parameters include a keepalive interval and a maximum number of keepalive retries. The VAM server assigns the configured keepalive parameters to clients in the ADVPN domain.

A client sends keepalives to the server at the specified interval. If a client does not receive any responses from the server after the maximum keepalive attempts (keepalive retries + 1), the client stops sending keepalives. If the VAM server does not receive any keepalives from a client before the timeout timer expires, the server removes information about the client and logs off the client. The timeout time is the product of the keepalive interval and keepalive attempts.

Restrictions and guidelines

Newly configured keepalive parameters do not affect registered VAM clients. They apply to subsequently registered clients.

If a device configured with dynamic NAT exists between the VAM server and VAM clients, configure the keepalive interval to be shorter than the aging time of NAT entries.

Configure proper values for the keepalive parameters depending on the network condition.

Procedure

1. Enter system view.

   system-view

2. Enter ADVPN domain view.

   vam server advpn-domain *domain-name* [**id** *domain-id*]

3. Configure keepalive parameters.

   keepalive interval *interval* **retry** *retries*

   By default, the keepalive interval is 180 seconds, and the maximum number of keepalive retries is 3.

Setting the retry timer

About this task

The VAM server starts the retry timer after it sends a request to a client. If the server does not receive a response from the client before the retry timer expires, the server resends the request. The server stops sending the request after receiving a response from the client or after the timeout timer (product of the keepalive interval and keepalive attempts) expires.

Procedure

1. Enter system view.

   system-view

2. Enter ADVPN domain view.

   vam server advpn-domain *domain-name* [**id** *domain-id*]

3. Set the retry timer.

   retry interval *interval*

   By default, the retry timer is 5 seconds.

Configuring the VAM client

VAM client tasks at a glance

To configure the VAM client, perform the following tasks:

1. Creating a VAM client
2. Enabling VAM clients
3. Specifying VAM servers
4. Specifying an ADVPN domain for a VAM client
5. Configuring a preshared key for a VAM client
6. (Optional.) Setting the retry interval and retry number for a VAM client
7. (Optional.) Setting the dumb timer for a VAM client
8. (Optional.) Configuring a username and password for a VAM client

Creating a VAM client

1. Enter system view.

   system-view
2. Create a VAM client and enter its view.

   vam client name *client-name*

Enabling VAM clients

1. Enter system view.

   system-view
2. Enable VAM clients. Choose one of the following tasks:
   - Enable one or all VAM clients.

     vam client enable [**name** *client-name*]
   - Execute the following commands in sequence to enable a VAM client:

     vam client name *client-name*

     client enable

   By default, no VAM clients are enabled.

Specifying VAM servers

About this task

You can specify a primary VAM server and a secondary VAM server for a VAM client. The client registers with both servers, and accepts settings from the server that first registers the client. When the server fails, the client uses the settings from the other server.

Restrictions and guidelines

If the specified primary and secondary VAM servers have the same address or name, only the primary VAM server takes effect.

The port number of a VAM server must be the same as that configured on the VAM server.

Procedure

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Specify the primary VAM server.

   server primary { **ip-address** *ip-address* | **ipv6-address** *ipv6-address* | **name** *host-name* } [**port** *port-number*]

   By default, no VAM server is specified.

4. (Optional.) Specify the secondary VAM server.

   server secondary { **ip-address** *ip-address* | **ipv6-address** *ipv6-address* | **name** *host-name* } [**port** *port-number*]

   By default, no VAM server is specified.

Specifying an ADVPN domain for a VAM client

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Specify an ADVPN domain for the VAM client.

   advpn-domain *domain-name*

   By default, no ADVPN domain is specified for a VAM client.

Configuring a preshared key for a VAM client

About this task

The preshared key is used to generate initial encryption and authentication keys during connection initialization. It is also used to generate encryption and authentication keys for subsequent packets if encryption and authentication are needed.

If a VAM client and the VAM server have different preshared keys, packet decryption and integrity verification will fail. The VAM client and server cannot establish a connection.

Restrictions and guidelines

The VAM server and the VAM clients in the same ADVPN domain must have the same preshared key.

Procedure

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Configure a preshared key for the VAM client.

   pre-shared-key { **cipher** | **simple** } *string*

   By default, no preshared key is configured for a VAM client.

Setting the retry interval and retry number for a VAM client

About this task

After a VAM client sends a request to the server, it resends the request if it does not receive any responses within the retry interval. If the client fails to receive a response after maximum attempts (retry times + 1), the client determines that the server is unreachable.

The *retry-times* setting does not apply to register and update requests. The client sends those requests at the retry interval until it goes offline.

Procedure

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Set the retry interval and retry number.

   retry interval *interval* **count** *retries*

   By default, the retry interval is 5 seconds, and the retry number is 3.

Setting the dumb timer for a VAM client

About this task

A VAM client starts the dumb timer after the timeout timer expires. The client does not process any packets during the dumb time. When the dumb timer expires, the client sends a new connection request to the VAM server.

Procedure

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Set the dumb timer.

   dumb-time *time-interval*

   By default, the dumb timer is 120 seconds.

Configuring a username and password for a VAM client

About this task

A VAM client uses its username and password for authentication on the VAM server.

Procedure

1. Enter system view.

   system-view

2. Enter VAM client view.

   vam client name *client-name*

3. Configure a username and password for the client.

   user *username* **password** { **cipher** | **simple** } *string*

   By default, no username and password are configured for a VAM client.

Configuring routing

ADVPN supports OSPF, RIP, and BGP for IPv4.

- When OSPF is used, set the network type of an OSPF interface to broadcast in a full-mesh network or to P2MP in a hub-spoke network. For more information about OSPF configuration, see *Layer 3—IP Routing Configuration Guide*.
- Full-mesh ADVPN does not support RIP. Only hub-spoke ADVPN supports RIP. When RIP is used in a hub-spoke network, you must disable split horizon. For more information about RIP configuration, see *Layer 3—IP Routing Configuration Guide*.
- When BGP is used, configure a routing policy to make sure the next hop of a route destined for a remote private network meets the following requirements:
  - In a full-mesh network, the next hop is the IP address of the peer spoke. EBGP does not support full-mesh.
  - In a hub-spoke network, the next hop is the IP address of the hub.

  For more information about BGP and routing policy configuration, see *Layer 3—IP Routing Configuration Guide*.

ADVPN supports OSPFv3, RIPng, and IPv6 BGP for IPv6.

- When OSPFv3 is used, set the network type of an OSPFv3 interface to broadcast in a full-mesh network or to P2MP in a hub-spoke network. For more information about OSPFv3 configuration, see *Layer 3—IP Routing Configuration Guide*.
- When RIPng is used, only the full-mesh network is supported. For more information about RIPng configuration, see *Layer 3—IP Routing Configuration Guide*.
- When IPv6 BGP is used, configure a routing policy to make sure the next hop of a route destined for a remote private network meets the following requirements:
  - In a full-mesh network, the next hop is the IP address of the peer spoke. EBGP does not support full-mesh.
  - In a hub-spoke network, the next hop is the IP address of the hub.

  For more information about IPv6 BGP and routing policy configuration, see *Layer 3—IP Routing Configuration Guide*.

Configuring an ADVPN tunnel interface

Restrictions and guidelines

ADVPN establishes tunnels over ADVPN tunnel interfaces. If multiple GRE ADVPN tunnel interfaces have the same source address or source interface, you must configure different GRE keys for the interfaces. For more information about GRE keys, see "Configuring GRE."

For more information about tunnel interface, see "Configuring tunneling." For more information about tunnel interface configuration commands, see tunneling commands in *VPN Command Reference.*

Procedure

1. Enter system view.

   system-view

2. Create an ADVPN tunnel interface and enter its view.

   interface tunnel *number* [**mode advpn** { **gre** | **udp** } [**ipv6**]]

   The two ends of an ADVPN tunnel must use the same tunnel mode.

3. Configure a private address for the tunnel interface.

   IPv4:

   ip address *ip-address* { *mask* | *mask-length* } [**sub**]

IPv6:

ipv6 address *ipv6-address prefix-length*

By default, no private address is configured for the tunnel interface.

All tunnel interfaces in a hub group must reside in the same private network.

4. Specify a source address or source interface for the tunnel interface.

   source { *ip-address* | *interface-type interface-number* }

   By default, no source address or source interface is configured for a tunnel interface.

   The specified source address or the IP address of the specified source interface is used as the source address of sent ADVPN packets.

5. (Optional.) Set the DF bit for ADVPN packets.

   tunnel dfbit enable

   By default, the DF bit is not set for ADVPN packets.

6. (Optional.) Set the source UDP port number of ADVPN packets.

   advpn source-port *port-number*

   By default, the source UDP port number of ADVPN packets is 18001.

   This command is available when the tunnel mode is UDP.

   If the **vam client** command configured on the tunnel interface has the **compatible** keyword, the tunnel interface must have a different source UDP port number than other tunnel interfaces.

7. Bind a VAM client to the tunnel interface.

   IPv4:

   vam client *client-name* [**compatible advpn0**]

   IPv6:

   vam ipv6 client *client-name*

   By default, no VAM client is bound to an ADVPN tunnel interface.

   A VAM client can be bound to only one IPv4 or IPv6 ADVPN tunnel interface.

8. (Optional.) Configure a private network for the tunnel interface.

   IPv4:

   advpn network *ip-address* { *mask-length* | *mask* } [**preference** *preference-value*]

   IPv6:

   advpn ipv6 network *prefix prefix-length* [**preference** *preference-value*]

   By default, no private network is configured for the tunnel interface.

   Set the preference of the private network route to be higher than other dynamic routing protocols, and lower than static routing.

9. (Optional.) Set the keepalive interval and the maximum number of keepalive attempts for the tunnel interface.

   keepalive interval *interval* **retry** *retries*

   By default, the keepalive interval is 180 seconds, and the maximum number of keepalive attempts is 3.

   The keepalive interval and the maximum number of keepalive attempts must be the same on the tunnel interfaces in an ADVPN domain.

10. (Optional.) Set the idle timeout time for the spoke-spoke tunnel.

    advpn session idle-time *time-interval*

    By default, the idle timeout time is 600 seconds.

The new idle timeout setting applies to both existing and subsequently established spoke-spoke tunnels.

11. (Optional.) Set the dumb timer for the tunnel interface.

    advpn session dumb-time *time-interval*

    By default, the dumb timer is 120 seconds.

    The new dumb timer setting only applies to subsequently established tunnels.

12. (Optional.) Configure an ADVPN group name.

    advpn group *group-name*

    By default, no ADVPN group name is configured.

    Perform this step on the spoke.

13. (Optional.) Configure a mapping between an ADVPN group and a QoS policy.

    advpn map group *group-name* **qos-policy** *policy-name* **outbound**

    By default, no ADVPN group-to-QoS policy mappings are configured.

    Perform this step on the hub.

Configuring IPsec for ADVPN tunnels

You can configure an IPsec profile to secure ADVPN tunnels:

1. Configure IPsec transform sets to specify the security protocols, authentication and encryption algorithms, and the encapsulation mode.
2. Configure an IKE-mode IPsec profile that uses the IPsec transform sets.
3. Apply the IPsec profile to an ADVPN tunnel interface.

For more information about IPsec configuration, see *Security Configuration Guide*.

Enabling ADVPN logging

About this task

This feature enables the device to generate logs for the ADVPN module and send the logs to the information center of the device. For the logs to be output correctly, you must also configure the information center on the device. For more information about information center configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable ADVPN logging.

   advpn logging enable

   By default, ADVPN logging is disabled.

Display and maintenance commands for ADVPN

△ **CAUTION:**

- The **reset vam server address-map** command clears IPv4 private-public address mapping information for VAM clients registered with the VAM server. When this command is executed, the system sends an error notification to VAM clients that have registered the private IPv4 addresses and logs off the clients.

- The **reset vam server ipv6 address-map** command clears IPv6 private-public address mapping information for VAM clients registered with the VAM server. When this command is executed, the system sends an error notification to VAM clients that have registered the private IPv6 addresses and logs off the clients.
- After you use the **reset vam client fsm** command to reset the FSM for a VAM client, the client will immediately try to come online.
- After you use the **reset vam client ipv6 fsm** command to reset the FSM for an IPv6 VAM client, the client will immediately try to come online.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPv4 private-to-public address mapping information for VAM clients registered with the VAM server.	**display vam server address-map** [**advpn-domain** *domain-name* [**private-address** *private-ip-address*]] [**verbose**]
Display IPv6 private-to-public address mapping information for VAM clients registered with the VAM server.	**display vam server ipv6 address-map** [**advpn-domain** *domain-name* [**private-address** *private-ipv6-address*]] [**verbose**]
Display IPv4 private networks for VAM clients registered with the VAM server.	**display vam server private-network** [**advpn-domain** *domain-name* [**private-address** *private-ip-address*]]
Display IPv6 private networks for VAM clients registered with the VAM server.	**display vam server ipv6 private-network** [**advpn-domain** *domain-name* [**private-address** *private-ipv6-address*]]
Display ADVPN domain statistics on the VAM server.	**display vam server statistics** [**advpn-domain** *domain-name*]
Display FSM information for VAM clients.	**display vam client fsm** [**name** *client-name*]
Display statistics for VAM clients.	**display vam client statistics** [**name** *client-name*]
Display IPv4 spoke-to-spoke tunnel establishment rules for VAM clients.	**display vam client shortcut interest** [**name** *client-name*]
Display IPv6 spoke-to-spoke tunnel establishment rules for VAM clients.	**display vam client shortcut ipv6 interest** [**name** *client-name*]
Display ADVPN group-to-QoS policy mappings.	**display advpn group-qos-map** [**interface tunnel** *number* [**group** *group-name*]]
Display IPv4 ADVPN tunnel information.	**display advpn session** [**interface tunnel** *number* [**private-address** *private-ip-address*]] [**verbose**]
Display IPv6 ADVPN tunnel information.	**display advpn ipv6 session** [**interface tunnel** *number* [**private-address** *private-ipv6-address*]] [**verbose**]
Display the number of ADVPN sessions in different states.	**display advpn session count**
Clear IPv4 private-to-public address	**reset vam server address-map** [**advpn-domain**

Task	Command
mapping information for VAM clients registered with the VAM server.	*domain-name* [**private-address** *private-ip-address*]]
Clear IPv6 private-to-public address mapping information for VAM clients registered with the VAM server.	**reset vam server ipv6 address-map** [**advpn-domain** *domain-name* [**private-address** *private-ipv6-address*]]
Clear ADVPN domain statistics on the VAM server.	**reset vam server statistics** [**advpn-domain** *domain-name*]
Reset the FSM for VAM clients.	**reset vam client** [**ipv6**] **fsm** [**name** *client-name*]
Clear statistics for VAM client.	**reset vam client statistics** [**name** *client-name*]
Delete IPv4 ADVPN tunnels.	**reset advpn session** [**interface tunnel** *number* [**private-address** *private-ip-address*]]
Delete IPv6 ADVPN tunnels.	**reset advpn ipv6 session** [**interface tunnel** *number* [**private-address** *private-ipv6-address*]]
Clear statistics for IPv4 ADVPN tunnels.	**reset advpn session statistics** [**interface tunnel** *number* [**private-address** *private-ip-address*]]
Clear statistics for IPv6 ADVPN tunnels.	**reset advpn ipv6 session statistics** [**interface tunnel** *number* [**private-address** *private-ipv6-address*]]

ADVPN configuration examples

Example: Configuring IPv4 full-mesh ADVPN

Network configuration

As shown in Figure 7, the primary and secondary VAM servers manage and maintain VAM client information for all hubs and spokes. The AAA server performs authentication and accounting for VAM clients. The two hubs back up each other, and perform data forwarding and route exchange.

- Establish a permanent ADVPN tunnel between each spoke and each hub.
- Establish a temporary ADVPN tunnel dynamically between the two spokes in the same ADVPN domain.

Figure 7 Network diagram

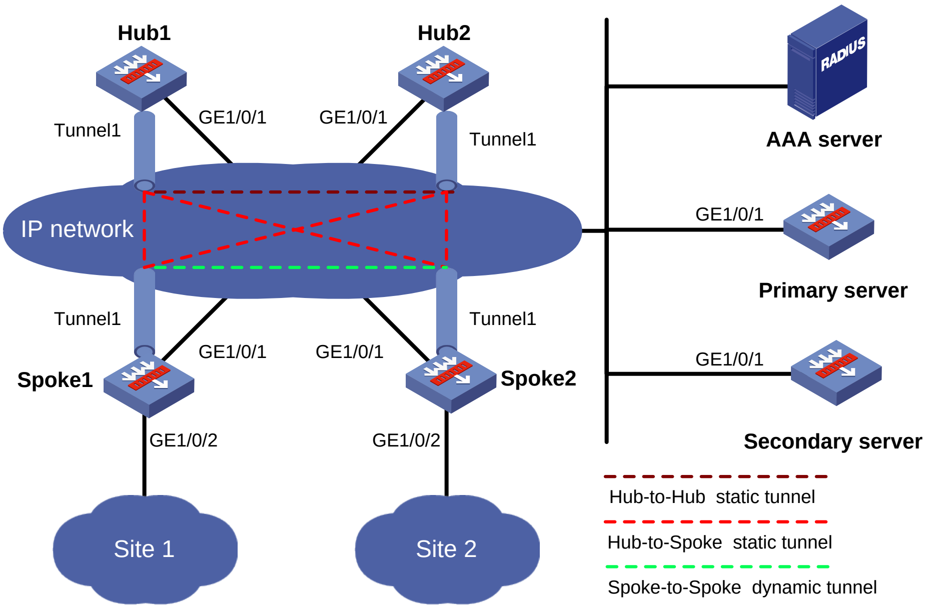


Table 1 Interface and IP address assignment

Device	Interface	IP address	Device	Interface	IP address
Hub 1	GE1/0/1	1.0.0.1/24	Spoke 1	GE1/0/1	1.0.0.3/24
	Tunnel1	192.168.0.1/24		GE1/0/2	192.168.1.1/24
Hub 2	GE1/0/1	1.0.0.2/24		Tunnel1	192.168.0.3/24
	Tunnel1	192.168.0.2/24	Spoke 2	GE1/0/1	1.0.0.4/24
AAA server		1.0.0.10/24		GE1/0/2	192.168.2.1/24
Primary server	GE1/0/1	1.0.0.11/24		Tunnel1	192.168.0.4/24
Secondary server	GE1/0/1	1.0.0.12/24			

Configuring the primary VAM server

1. Assign an IP address to GigabitEthernet 1/0/1.

```
<PrimaryServer> system-view
[PrimaryServer] interface gigabitethernet 1/0/1
[PrimaryServer-GigabitEthernet1/0/1] ip address 1.0.0.11 255.255.255.0
[PrimaryServer-GigabitEthernet1/0/1] quit
```

2. Add the interface to a security zone.

```
[PrimaryServer] security-zone name untrust
[PrimaryServer-security-zone-Untrust] import interface gigabitethernet 1/0/1
[PrimaryServer-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow the VAM server to send VAM protocol packets to the VAM clients.

```
[PrimaryServer] security-policy ip
[PrimaryServer-security-policy-ip] rule name advpnlocalout
[PrimaryServer-security-policy-ip-1-advpnlocalout] source-zone local
[PrimaryServer-security-policy-ip-1-advpnlocalout] destination-zone untrust
```

```
[PrimaryServer-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.11
[PrimaryServer-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.1
1.0.0.12
[PrimaryServer-security-policy-ip-1-ipseclocalout] action pass
[PrimaryServer-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow the VAM server to receive VAM protocol packets from the VAM clients.

```
[PrimaryServer-security-policy-ip] rule name advpnlocalin
[PrimaryServer-security-policy-ip-2-advpnlocalin] source-zone untrust
[PrimaryServer-security-policy-ip-2-advpnlocalin] destination-zone local
[PrimaryServer-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.1 1.0.0.12
[PrimaryServer-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.11
[PrimaryServer-security-policy-ip-2-advpnlocalin] action pass
[PrimaryServer-security-policy-ip-2-advpnlocalin] quit
[PrimaryServer-security-policy-ip] quit
```

4. Configure AAA:

# Configure RADIUS scheme **abc**.

```
[PrimaryServer] radius scheme abc
[PrimaryServer-radius-abc] primary authentication 1.0.0.10 1812
[PrimaryServer-radius-abc] primary accounting 1.0.0.10 1813
[PrimaryServer-radius-abc] key authentication simple 123
[PrimaryServer-radius-abc] key accounting simple 123
[PrimaryServer-radius-abc] user-name-format without-domain
[PrimaryServer-radius-abc] quit
[PrimaryServer] radius session-control enable
```

# Configure AAA methods for ISP domain **abc**.

```
[PrimaryServer] domain abc
[PrimaryServer-isp-abc] authentication advpn radius-scheme abc
[PrimaryServer-isp-abc] accounting advpn radius-scheme abc
[PrimaryServer-isp-abc] quit
[PrimaryServer] domain default enable abc
```

5. Configure the VAM server:

# Create ADVPN domain **abc**.

```
[PrimaryServer] vam server advpn-domain abc id 1
```

# Create hub group 0.

```
[PrimaryServer-vam-server-domain-abc] hub-group 0
```

# Specify hub private IPv4 addresses.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub private-address 192.168.0.1
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub private-address 192.168.0.2
```

# Specify a spoke private IPv4 network.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] spoke private-address network
192.168.0.0 255.255.255.0
[PrimaryServer-vam-server-domain-abc-hub-group-0] quit
```

# Set the preshared key to **123456**.

```
[PrimaryServer-vam-server-domain-abc] pre-shared-key simple 123456
```

# Set the authentication mode to CHAP.

```
[PrimaryServer-vam-server-domain-abc] authentication-method chap
```

# Enable the VAM server for the ADVPN domain.

```
[PrimaryServer-vam-server-domain-abc] server enable
[PrimaryServer-vam-server-domain-abc] quit
```

Configuring the secondary VAM server

# Configure the secondary VAM server in the same way that the primary server is configured. (Details not shown.)

Configuring Hub 1

1. Assign IP addresses to interfaces:

   # Assign an IPv4 address to GigabitEthernet 1/0/1.

   ```
   <Hub1> system-view
   [Hub1] interface gigabitethernet 1/0/1
   [Hub1-GigabitEthernet1/0/1] ip address 1.0.0.1 255.255.255.0
   [Hub1-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

   ```
   [Hub1] interface tunnel1 mode advpn gre
   [Hub1-Tunnel1] ip address 192.168.0.1 255.255.255.0
   ```

2. Add the interfaces to a security zone.

   ```
   [Hub1] security-zone name untrust
   [Hub1-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Hub1-security-zone-Untrust] import interface tunnel1
   [Hub1-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Hub 1 to send VAM protocol packets to the VAM server.

   ```
   [Hub1] security-policy ip
   [Hub1-security-policy-ip] rule name advpnlocalout
   [Hub1-security-policy-ip-1-advpnlocalout] source-zone local
   [Hub1-security-policy-ip-1-advpnlocalout] destination-zone untrust
   [Hub1-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.1
   [Hub1-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
   [Hub1-security-policy-ip-1-ipseclocalout] action pass
   [Hub1-security-policy-ip-1-ipseclocalout] quit
   ```

   # Configure a rule named **advpnlocalin** to allow Hub 1 to receive VAM protocol packets from the VAM server.

   ```
   [Hub1-security-policy-ip] rule name advpnlocalin
   [Hub1-security-policy-ip-2-advpnlocalin] source-zone untrust
   [Hub1-security-policy-ip-2-advpnlocalin] destination-zone local
   [Hub1-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
   [Hub1-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.1
   [Hub1-security-policy-ip-2-advpnlocalin] action pass
   [Hub1-security-policy-ip-2-advpnlocalin] quit
   ```

   # Configure a rule named **ipseclocalout** to allow Hub 1 to send IPsec negotiation packets to other VAM clients.

   ```
   [Hub1-security-policy-ip] rule name ipseclocalout
   [Hub1-security-policy-ip-3-ipseclocalout] source-zone local
   [Hub1-security-policy-ip-3-ipseclocalout] destination-zone untrust
   [Hub1-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.1
   ```

```
[Hub1-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.2 1.0.0.4
[Hub1-security-policy-ip-3-ipseclocalout] action pass
[Hub1-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 1 to receive IPsec negotiation packets from other VAM clients.

```
[Hub1-security-policy-ip] rule name ipseclocalin
[Hub1-security-policy-ip-4-ipseclocalin] source-zone untrust
[Hub1-security-policy-ip-4-ipseclocalin] destination-zone local
[Hub1-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.2 1.0.0.4
[Hub1-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.1
[Hub1-security-policy-ip-4-ipseclocalin] action pass
[Hub1-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 1 to send OSPF packets to other VAM clients.

```
[Hub1-security-policy-ip] rule name ospflocalout
[Hub1-security-policy-ip-5-ospflocalout] source-zone local
[Hub1-security-policy-ip-5-ospflocalout] destination-zone untrust
[Hub1-security-policy-ip-5-ospflocalout] service ospf
[Hub1-security-policy-ip-5-ospflocalout] action pass
[Hub1-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 1 to receive OSPF packets from other VAM clients.

```
[Hub1-security-policy-ip] rule name ospflocalin
[Hub1-security-policy-ip-6-ospflocalin] source-zone untrust
[Hub1-security-policy-ip-6-ospflocalin] destination-zone local
[Hub1-security-policy-ip-6-ospflocalin] service ospf
[Hub1-security-policy-ip-6-ospflocalin] action pass
[Hub1-security-policy-ip-6-ospflocalin] quit
```

4. Configure the VAM client:

# Create VAM client **Hub1**.

```
[Hub1] vam client name Hub1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub1-vam-client-Hub1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub1-vam-client-Hub1] pre-shared-key simple 123456
```

# Set both the username and password to **hub1**.

```
[Hub1-vam-client-Hub1] user hub1 password simple hub1
```

# Specify the primary and secondary VAM servers.

```
[Hub1-vam-client-Hub1] server primary ip-address 1.0.0.11
[Hub1-vam-client-Hub1] server secondary ip-address 1.0.0.12
```

# Enable the VAM client.

```
[Hub1-vam-client-Hub1] client enable
[Hub1-vam-client-Hub1] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Hub1] ike keychain abc
[Hub1-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
[Hub1-ike-keychain-abc] quit
```

```
[Hub1] ike profile abc
[Hub1-ike-profile-abc] keychain abc
[Hub1-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Hub1] ipsec transform-set abc
[Hub1-ipsec-transform-set-abc] encapsulation-mode transport
[Hub1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Hub1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Hub1-ipsec-transform-set-abc] quit
[Hub1] ipsec profile abc isakmp
[Hub1-ipsec-profile-isakmp-abc] transform-set abc
[Hub1-ipsec-profile-isakmp-abc] ike-profile abc
[Hub1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise the private network.

```
[Hub1] ospf 1
[Hub1-ospf-1] area 0
[Hub1-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[Hub1-ospf-1-area-0.0.0.0] quit
[Hub1-ospf-1] quit
```

7. Configure interface Tunnel 1.

```
[Hub1] interface tunnel1
[Hub1-Tunnel1] vam client Hub1
[Hub1-Tunnel1] ospf network-type broadcast
[Hub1-Tunnel1] source gigabitethernet 1/0/1
[Hub1-Tunnel1] tunnel protection ipsec profile abc
[Hub1-Tunnel1] quit
```

Configuring Hub 2

1. Assign IP addresses to interfaces:

# Assign an IPv4 address to GigabitEthernet 1/0/1.

```
<Hub2> system-view
[Hub2] interface gigabitethernet 1/0/1
[Hub2-GigabitEthernet1/0/1] ip address 1.0.0.2 255.255.255.0
[Hub2-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

```
[Hub2] interface tunnel1 mode advpn gre
[Hub2-Tunnel1] ip address 192.168.0.2 255.255.255.0
```

2. Add the interfaces to a security zone.

```
[Hub2] security-zone name untrust
[Hub2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Hub2-security-zone-Untrust] import interface tunnel1
[Hub2-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Hub 2 to send VAM protocol packets to the VAM server.

```
[Hub2] security-policy ip
[Hub2-security-policy-ip] rule name advpnlocalout
```

```
[Hub2-security-policy-ip-1-advpnlocalout] source-zone local
[Hub2-security-policy-ip-1-advpnlocalout] destination-zone untrust
[Hub2-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.2
[Hub2-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
[Hub2-security-policy-ip-1-ipseclocalout] action pass
[Hub2-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Hub 2 to receive VAM protocol packets from the VAM server.

```
[Hub2-security-policy-ip] rule name advpnlocalin
[Hub2-security-policy-ip-2-advpnlocalin] source-zone untrust
[Hub2-security-policy-ip-2-advpnlocalin] destination-zone local
[Hub2-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
[Hub2-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.2
[Hub2-security-policy-ip-2-advpnlocalin] action pass
[Hub2-security-policy-ip-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Hub 2 to send IPsec negotiation packets to other VAM clients.

```
[Hub2-security-policy-ip] rule name ipseclocalout
[Hub2-security-policy-ip-3-ipseclocalout] source-zone local
[Hub2-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Hub2-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.2
[Hub2-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.4
[Hub2-security-policy-ip-3-ipseclocalout] action pass
[Hub2-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 2 to receive IPsec negotiation packets from other VAM clients.

```
[Hub2-security-policy-ip] rule name ipseclocalin
[Hub2-security-policy-ip-4-ipseclocalin] source-zone untrust
[Hub2-security-policy-ip-4-ipseclocalin] destination-zone local
[Hub2-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.4
[Hub2-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.2
[Hub2-security-policy-ip-4-ipseclocalin] action pass
[Hub2-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 2 to send OSPF packets to other VAM clients.

```
[Hub2-security-policy-ip] rule name ospflocalout
[Hub2-security-policy-ip-5-ospflocalout] source-zone local
[Hub2-security-policy-ip-5-ospflocalout] destination-zone untrust
[Hub2-security-policy-ip-5-ospflocalout] service ospf
[Hub2-security-policy-ip-5-ospflocalout] action pass
[Hub2-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 2 to receive OSPF packets from other VAM clients.

```
[Hub2-security-policy-ip] rule name ospflocalin
[Hub2-security-policy-ip-6-ospflocalin] source-zone untrust
[Hub2-security-policy-ip-6-ospflocalin] destination-zone local
[Hub2-security-policy-ip-6-ospflocalin] service ospf
[Hub2-security-policy-ip-6-ospflocalin] action pass
[Hub2-security-policy-ip-6-ospflocalin] quit
```

```
[Hub2-security-policy-ip] quit
```

4. Configure the VAM client:

   # Create VAM client **Hub2**.

   ```
   <Hub2> system-view
   [Hub2] vam client name Hub2
   ```

   # Specify ADVPN domain **abc** for the VAM client.

   ```
   [Hub2-vam-client-Hub2] advpn-domain abc
   ```

   # Set the preshared key to **123456**.

   ```
   [Hub2-vam-client-Hub2] pre-shared-key simple 123456
   ```

   # Set both the username and password to **hub2**.

   ```
   [Hub2-vam-client-Hub2] user hub2 password simple hub2
   ```

   # Specify the primary and secondary VAM servers.

   ```
   [Hub2-vam-client-Hub2] server primary ip-address 1.0.0.11
   [Hub2-vam-client-Hub2] server secondary ip-address 1.0.0.12
   ```

   # Enable the VAM client.

   ```
   [Hub2-vam-client-Hub2] client enable
   [Hub2-vam-client-Hub2] quit
   ```

5. Configure an IPsec profile:

   # Configure IKE.

   ```
   [Hub2] ike keychain abc
   [Hub2-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
   [Hub2-ike-keychain-abc] quit
   [Hub2] ike profile abc
   [Hub2-ike-profile-abc] keychain abc
   [Hub2-ike-profile-abc] quit
   ```

   # Configure the IPsec profile.

   ```
   [Hub2] ipsec transform-set abc
   [Hub2-ipsec-transform-set-abc] encapsulation-mode transport
   [Hub2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
   [Hub2-ipsec-transform-set-abc] esp authentication-algorithm sha1
   [Hub2-ipsec-transform-set-abc] quit
   [Hub2] ipsec profile abc isakmp
   [Hub2-ipsec-profile-isakmp-abc] transform-set abc
   [Hub2-ipsec-profile-isakmp-abc] ike-profile abc
   [Hub2-ipsec-profile-isakmp-abc] quit
   ```

6. Configure OSPF to advertise the private network.

   ```
   [Hub2] ospf 1
   [Hub2-ospf-1] area 0
   [Hub2-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
   [Hub2-ospf-1-area-0.0.0.0] quit
   [Hub2-ospf-1] quit
   ```

7. Configure interface Tunnel 1.

   ```
   [Hub2] interface tunnel1
   [Hub2-Tunnel1] vam client Hub2
   [Hub2-Tunnel1] ospf network-type broadcast
   [Hub2-Tunnel1] source gigabitethernet 1/0/1
   [Hub2-Tunnel1] tunnel protection ipsec profile abc
   ```

```
[Hub2-Tunnel1] quit
```

Configuring Spoke 1

1. Assign IP addresses to interfaces:

   # Assign an IPv4 address to interface GigabitEthernet 1/0/1.

   ```
   <Spoke1> system-view
   [Spoke1] interface gigabitethernet 1/0/1
   [Spoke1-GigabitEthernet1/0/1] ip address 1.0.0.3 255.255.255.0
   [Spoke1-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

   ```
   [Spoke1] interface tunnel 1 mode advpn gre
   [Spoke1-Tunnel1] ip address 192.168.0.3 255.255.255.0
   ```

   # Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

   ```
   [Spoke1] security-zone name untrust
   [Spoke1-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Spoke1-security-zone-Untrust] import interface tunnel1
   [Spoke1-security-zone-Untrust] quit
   [Spoke1] security-zone name trust
   [Spoke1-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Spoke1-security-zone-Trust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Spoke 1 to send VAM protocol packets to the VAM server.

   ```
   [Spoke1] security-policy ip
   [Spoke1-security-policy-ip] rule name advpnlocalout
   [Spoke1-security-policy-ip-1-advpnlocalout] source-zone local
   [Spoke1-security-policy-ip-1-advpnlocalout] destination-zone untrust
   [Spoke1-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.3
   [Spoke1-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
   [Spoke1-security-policy-ip-1-ipseclocalout] action pass
   [Spoke1-security-policy-ip-1-ipseclocalout] quit
   ```

   # Configure a rule named **advpnlocalin** to allow Spoke 1 to receive VAM protocol packets from the VAM server.

   ```
   [Spoke1-security-policy-ip] rule name advpnlocalin
   [Spoke1-security-policy-ip-2-advpnlocalin] source-zone untrust
   [Spoke1-security-policy-ip-2-advpnlocalin] destination-zone local
   [Spoke1-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
   [Spoke1-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.3
   [Spoke1-security-policy-ip-2-advpnlocalin] action pass
   [Spoke1-security-policy-ip-2-advpnlocalin] quit
   ```

   # Configure a rule named **ipseclocalout** to allow Spoke 1 to send IPsec negotiation packets to other VAM clients.

   ```
   [Spoke1-security-policy-ip] rule name ipseclocalout
   [Spoke1-security-policy-ip-3-ipseclocalout] source-zone local
   [Spoke1-security-policy-ip-3-ipseclocalout] destination-zone untrust
   [Spoke1-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.3
   [Spoke1-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.4
   ```

```
[Spoke1-security-policy-ip-3-ipseclocalout] action pass
[Spoke1-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 1 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke1-security-policy-ip] rule name ipseclocalin
[Spoke1-security-policy-ip-4-ipseclocalin] source-zone untrust
[Spoke1-security-policy-ip-4-ipseclocalin] destination-zone local
[Spoke1-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.4
[Spoke1-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.3
[Spoke1-security-policy-ip-4-ipseclocalin] action pass
[Spoke1-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 1 to send OSPF packets to other VAM clients.

```
[Spoke1-security-policy-ip] rule name ospflocalout
[Spoke1-security-policy-ip-5-ospflocalout] source-zone local
[Spoke1-security-policy-ip-5-ospflocalout] destination-zone untrust
[Spoke1-security-policy-ip-5-ospflocalout] service ospf
[Spoke1-security-policy-ip-5-ospflocalout] action pass
[Spoke1-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 1 to receive OSPF packets from other VAM clients.

```
[Spoke1-security-policy-ip] rule name ospflocalin
[Spoke1-security-policy-ip-6-ospflocalin] source-zone untrust
[Spoke1-security-policy-ip-6-ospflocalin] destination-zone local
[Spoke1-security-policy-ip-6-ospflocalin] service ospf
[Spoke1-security-policy-ip-6-ospflocalin] action pass
[Spoke1-security-policy-ip-6-ospflocalin] quit
[Spoke1-security-policy-ip] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke1**.

```
<Spoke1> system-view
[Spoke1] vam client name Spoke1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke1-vam-client-Spoke1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke1-vam-client-Spoke1] pre-shared-key simple 123456
```

# Set both the username and password to **spoke1**.

```
[Spoke1-vam-client-Spoke1] user spoke1 password simple spoke1
```

# Specify the primary and secondary VAM servers.

```
[Spoke1-vam-client-Spoke1] server primary ip-address 1.0.0.11
[Spoke1-vam-client-Spoke1] server secondary ip-address 1.0.0.12
```

# Enable the VAM client.

```
[Spoke1-vam-client-Spoke1] client enable
[Spoke1-vam-client-Spoke1] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Spoke1] ike keychain abc
[Spoke1-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
```

```
[Spoke1-ike-keychain-abc] quit
[Spoke1] ike profile abc
[Spoke1-ike-profile-abc] keychain abc
[Spoke1-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Spoke1] ipsec transform-set abc
[Spoke1-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke1-ipsec-transform-set-abc] quit
[Spoke1] ipsec profile abc isakmp
[Spoke1-ipsec-profile-isakmp-abc] transform-set abc
[Spoke1-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise private networks.

```
[Spoke1] ospf 1
[Spoke1-ospf-1] area 0
[Spoke1-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[Spoke1-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[Spoke1-ospf-1-area-0.0.0.0] quit
[Spoke1-ospf-1] quit
```

7. Configure interface Tunnel 1. Set its DR priority to 0 to prevent Spoke 1 from participating in DR/BDR election.

```
[Spoke1] interface tunnel1
[Spoke1-Tunnel1] vam client Spoke1
[Spoke1-Tunnel1] ospf network-type broadcast
[Spoke1-Tunnel1] ospf dr-priority 0
[Spoke1-Tunnel1] source gigabitethernet 1/0/1
[Spoke1-Tunnel1] tunnel protection ipsec profile abc
[Spoke1-Tunnel1] quit
```

Configuring Spoke 2

1. Assign IP addresses to interfaces:

# Assign an IPv4 address to interface GigabitEthernet 1/0/1.

```
<Spoke2> system-view
[Spoke2] interface gigabitethernet 1/0/1
[Spoke2-GigabitEthernet1/0/1] ip address 1.0.0.4 255.255.255.0
[Spoke2-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

```
[Spoke2] interface tunnel 1 mode advpn gre
[Spoke2-Tunnel1] ip address 192.168.0.4 255.255.255.0
```

# Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke2] security-zone name untrust
[Spoke2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Spoke2-security-zone-Untrust] import interface tunnel1
[Spoke2-security-zone-Untrust] quit
```

```
[Spoke2] security-zone name trust
[Spoke2-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke2-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Spoke 2 to send VAM protocol packets to the VAM server.

```
[Spoke2] security-policy ip
[Spoke2-security-policy-ip] rule name advpnlocalout
[Spoke2-security-policy-ip-1-advpnlocalout] source-zone local
[Spoke2-security-policy-ip-1-advpnlocalout] destination-zone untrust
[Spoke2-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.4
[Spoke2-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
[Spoke2-security-policy-ip-1-ipseclocalout] action pass
[Spoke2-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Spoke 2 to receive VAM protocol packets from the VAM server.

```
[Spoke2-security-policy-ip] rule name advpnlocalin
[Spoke2-security-policy-ip-2-advpnlocalin] source-zone untrust
[Spoke2-security-policy-ip-2-advpnlocalin] destination-zone local
[Spoke2-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
[Spoke2-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.4
[Spoke2-security-policy-ip-2-advpnlocalin] action pass
[Spoke2-security-policy-ip-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 2 to send IPsec negotiation packets to other VAM clients.

```
[Spoke2-security-policy-ip] rule name ipseclocalout
[Spoke2-security-policy-ip-3-ipseclocalout] source-zone local
[Spoke2-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Spoke2-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.4
[Spoke2-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.3
[Spoke2-security-policy-ip-3-ipseclocalout] action pass
[Spoke2-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 2 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke2-security-policy-ip] rule name ipseclocalin
[Spoke2-security-policy-ip-4-ipseclocalin] source-zone untrust
[Spoke2-security-policy-ip-4-ipseclocalin] destination-zone local
[Spoke2-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.3
[Spoke2-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.4
[Spoke2-security-policy-ip-4-ipseclocalin] action pass
[Spoke2-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 2 to send OSPF packets to other VAM clients.

```
[Spoke2-security-policy-ip] rule name ospflocalout
[Spoke2-security-policy-ip-5-ospflocalout] source-zone local
[Spoke2-security-policy-ip-5-ospflocalout] destination-zone untrust
[Spoke2-security-policy-ip-5-ospflocalout] service ospf
[Spoke2-security-policy-ip-5-ospflocalout] action pass
[Spoke2-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 2 to receive OSPF packets from other VAM clients.

```
[Spoke2-security-policy-ip] rule name ospflocalin
[Spoke2-security-policy-ip-6-ospflocalin] source-zone untrust
[Spoke2-security-policy-ip-6-ospflocalin] destination-zone local
[Spoke2-security-policy-ip-6-ospflocalin] service ospf
[Spoke2-security-policy-ip-6-ospflocalin] action pass
[Spoke2-security-policy-ip-6-ospflocalin] quit
[Spoke2-security-policy-ip] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke2**.

```
<Spoke2> system-view
[Spoke2] vam client name Spoke2
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke2-vam-client-Spoke2] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke2-vam-client-Spoke2] pre-shared-key simple 123456
```

# Set both the username and password to **spoke2**.

```
[Spoke2-vam-client-Spoke2] user spoke2 password simple spoke2
```

# Specify the primary and secondary VAM servers.

```
[Spoke2-vam-client-Spoke2] server primary ip-address 1.0.0.11
[Spoke2-vam-client-Spoke2] server secondary ip-address 1.0.0.12
```

# Enable the VAM client.

```
[Spoke2-vam-client-Spoke2] client enable
[Spoke2-vam-client-Spoke2] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Spoke2] ike keychain abc
[Spoke2-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
[Spoke2-ike-keychain-abc] quit
[Spoke2] ike profile abc
[Spoke2-ike-profile-abc] keychain abc
[Spoke2-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Spoke2] ipsec transform-set abc
[Spoke2-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke2-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke2-ipsec-transform-set-abc] quit
[Spoke2] ipsec profile abc isakmp
[Spoke2-ipsec-profile-isakmp-abc] transform-set abc
[Spoke2-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke2-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise private networks.

```
[Spoke2] ospf 1
[Spoke2-ospf-1] area 0
[Spoke2-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
```

```
[Spoke2-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[Spoke2-ospf-1-area-0.0.0.0] quit
[Spoke2-ospf-1] quit
```

7. Configure interface Tunnel 1. Set its DR priority to 0 to prevent Spoke 2 from participating in DR/BDR election.

```
[Spoke2] interface tunnel1
[Spoke2-Tunnel1] vam client Spoke2
[Spoke2-Tunnel1] ospf network-type broadcast
[Spoke2-Tunnel1] ospf dr-priority 0
[Spoke2-Tunnel1] source gigabitethernet 1/0/1
[Spoke2-Tunnel1] tunnel protection ipsec profile abc
[Spoke2-Tunnel1] quit
```

Verifying the configuration

# Display IPv4 address mapping information for all VAM clients registered with the primary VAM server.

```
[PrimaryServer] display vam server address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address  Public address               Type   NAT  Holding time
0          192.168.0.1      1.0.0.1                      Hub    No   0H 52M  7S
0          192.168.0.2      1.0.0.2                      Hub    No   0H 47M 31S
0          192.168.0.3      1.0.0.3                      Spoke  No   0H 28M 25S
0          192.168.0.4      1.0.0.4                      Spoke  No   0H 19M 15S
```

# Display IPv4 address mapping information for all VAM clients registered with the secondary VAM server.

```
[SecondaryServer] display vam server address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address  Public address               Type   NAT  Holding time
0          192.168.0.1      1.0.0.1                      Hub    No   0H 52M  7S
0          192.168.0.2      1.0.0.2                      Hub    No   0H 47M 31S
0          192.168.0.3      1.0.0.3                      Spoke  No   0H 28M 25S
0          192.168.0.4      1.0.0.4                      Spoke  No   0H 19M 15S
```

The output shows that Hub 1, Hub 2, Spoke 1, and Spoke 2 all have registered their address mapping information with the VAM servers.

# Display IPv4 ADVPN tunnel information on Hubs. This example uses Hub 1.

```
[Hub1] display advpn session
Interface         : Tunnel1
Number of sessions: 3
Private address  Public address              Port  Type  State      Holding time
192.168.0.2      1.0.0.2                     --    H-H   Success    0H 46M  8S
192.168.0.3      1.0.0.3                     --    H-S   Success    0H 27M 27S
192.168.0.4      1.0.0.4                     --    H-S   Success    0H 18M 18S
```

The output shows that Hub 1 has established a permanent tunnel to Hub 2, Spoke 1, and Spoke 2.

# Display IPv4 ADVPN tunnel information on Spokes. This example uses Spoke 1.

```
[Spoke1] display advpn session
Interface         : Tunnel1
```

```
Number of sessions: 2
Private address  Public address              Port  Type  State      Holding time
192.168.0.1      1.0.0.1                     --    S-H   Success    0H 46M  8S
192.168.0.2      1.0.0.2                     --    S-H   Success    0H 46M  8S
```

The output shows that Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.

# Verify that Spoke 1 can ping the private address 192.168.0.4 of Spoke 2.

```
[Spoke1] ping 192.168.0.4
Ping 192.168.0.4 (192.168.0.4): 56 data bytes, press CTRL_C to break
56 bytes from 192.168.0.4: icmp_seq=0 ttl=255 time=4.000 ms
56 bytes from 192.168.0.4: icmp_seq=1 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 192.168.0.4 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.000/4.000/1.549 ms
```

# Display IPv4 ADVPN tunnel information on Spokes. This example uses Spoke 1.

```
[Spoke1] display advpn session
Interface         : Tunnel1
Number of sessions: 3
Private address  Public address              Port  Type  State      Holding time
192.168.0.1      1.0.0.1                     --    S-H   Success    0H 46M  8S
192.168.0.2      1.0.0.2                     --    S-H   Success    0H 46M  8S
192.168.0.4      1.0.0.4                     --    S-S   Success    0H  0M  1S
```

The output shows the following information:

- Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.
- Spoke 1 has established a temporary spoke-spoke tunnel to Spoke 2.

Example: Configuring IPv6 full-mesh ADVPN

Network configuration

As shown in Figure 8, the primary and secondary VAM servers manage and maintain VAM client information for all hubs and spokes. The AAA server performs authentication and accounting for VAM clients. The two hubs back up each other, and perform data forwarding and route exchange.

- Establish a permanent ADVPN tunnel between each spoke and each hub.
- Establish a temporary ADVPN tunnel dynamically between the two spokes in the same ADVPN domain.

Figure 8 Network diagram

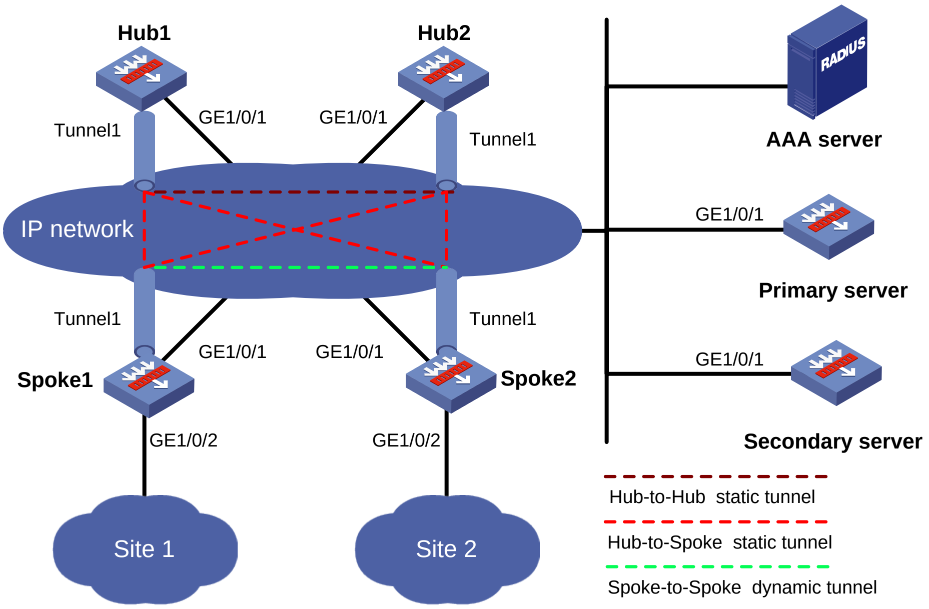


Table 2 Interface and IP address assignment

Device	Interface	IP address	Device	Interface	IP address
Hub 1	GE1/0/1	1::1/64	Spoke 1	GE1/0/1	1::3/64
	Tunnel1	192:168::1/64		GE1/0/2	192:168:1::1/64
Hub 2	GE1/0/1	1::2/64		Tunnel1	192:168::3/64
	Tunnel1	192:168::2/64	Spoke 2	GE1/0/1	1::4/64
AAA server		1::10/64		GE1/0/2	192:168:2::1/64
Primary server	GE1/0/1	1::11/64		Tunnel1	192:168::4/64
Secondary server	GE1/0/1	1::12/64			

Configuring the primary VAM server

1. Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

```
<PrimaryServer> system-view
[PrimaryServer] interface gigabitethernet 1/0/1
[PrimaryServer-GigabitEthernet1/0/1] ipv6 address 1::11/64
[PrimaryServer-GigabitEthernet1/0/1] quit
```

2. Add the interface to a security zone.

```
[PrimaryServer] security-zone name untrust
[PrimaryServer-security-zone-Untrust] import interface gigabitethernet 1/0/1
[PrimaryServer-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow the VAM server to send VAM protocol packets to the VAM clients.

```
[PrimaryServer] security-policy ipv6
[PrimaryServer-security-policy-ipv6] rule name advpnlocalout
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] source-zone local
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
```

```
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::11
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::1 1::12
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] action pass
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] quit
```

# Configure a rule named **advpnlocalin** to allow the VAM server to receive VAM protocol packets from the VAM clients.

```
[PrimaryServer-security-policy-ipv6] rule name advpnlocalin
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] destination-zone local
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::1 1::12
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::11
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] action pass
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] quit
[PrimaryServer-security-policy-ipv6] quit
```

4. Configure AAA:

# Configure RADIUS scheme **abc**.

```
<PrimaryServer> system-view
[PrimaryServer] radius scheme abc
[PrimaryServer-radius-abc] primary authentication ipv6 1::10 1812
[PrimaryServer-radius-abc] primary accounting ipv6 1::10 1813
[PrimaryServer-radius-abc] key authentication simple 123
[PrimaryServer-radius-abc] key accounting simple 123
[PrimaryServer-radius-abc] user-name-format without-domain
[PrimaryServer-radius-abc] quit
[PrimaryServer] radius session-control enable
```

# Configure AAA methods for ISP domain **abc**.

```
[PrimaryServer] domain abc
[PrimaryServer-isp-abc] authentication advpn radius-scheme abc
[PrimaryServer-isp-abc] accounting advpn radius-scheme abc
[PrimaryServer-isp-abc] quit
[PrimaryServer] domain default enable abc
```

5. Configure the VAM server:

# Create ADVPN domain **abc**.

```
[PrimaryServer] vam server advpn-domain abc id 1
```

# Create hub group 0.

```
[PrimaryServer-vam-server-domain-abc] hub-group 0
```

# Specify hub private IPv6 addresses.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub ipv6 private-address 192:168::1
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub ipv6 private-address 192:168::2
```

# Specify a spoke private IPv6 network.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] spoke ipv6 private-address network 192:168::0 64
```

# Set the preshared key to **123456**.

```
[PrimaryServer-vam-server-domain-abc] pre-shared-key simple 123456
```

# Set the authentication mode to CHAP.

```
[PrimaryServer-vam-server-domain-abc] authentication-method chap
```

# Enable the VAM server for the ADVPN domain.

```
[PrimaryServer-vam-server-domain-abc] server enable
[PrimaryServer-vam-server-domain-abc] quit
```

Configuring the secondary VAM server

# Configure the secondary VAM server in the same way that the primary server is configured. (Details not shown.)

Configuring Hub 1

1. Assign IP addresses to interfaces:

   # Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

   ```
   <Hub1> system-view
   [Hub1] interface gigabitethernet 1/0/1
   [Hub1-GigabitEthernet1/0/1] ipv6 address 1::1 64
   [Hub1-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

   ```
   [Hub1] interface tunnel 1 mode advpn gre ipv6
   [Hub1-Tunnel1] ipv6 address 192:168::1 64
   [Hub1-Tunnel1] ipv6 address fe80::1 link-local
   ```

2. Add the interfaces to a security zone.

   ```
   [Hub1] security-zone name untrust
   [Hub1-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Hub1-security-zone-Untrust] import interface tunnel1
   [Hub1-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Hub 1 to send VAM protocol packets to the VAM server.

   ```
   [Hub1] security-policy ipv6
   [Hub1-security-policy-ipv6] rule name advpnlocalout
   [Hub1-security-policy-ipv6-1-advpnlocalout] source-zone local
   [Hub1-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
   [Hub1-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::1
   [Hub1-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
   [Hub1-security-policy-ipv6-1-ipseclocalout] action pass
   [Hub1-security-policy-ipv6-1-ipseclocalout] quit
   ```

   # Configure a rule named **advpnlocalin** to allow Hub 1 to receive VAM protocol packets from the VAM server.

   ```
   [Hub1-security-policy-ipv6] rule name advpnlocalin
   [Hub1-security-policy-ipv6-2-advpnlocalin] source-zone untrust
   [Hub1-security-policy-ipv6-2-advpnlocalin] destination-zone local
   [Hub1-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
   [Hub1-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::1
   [Hub1-security-policy-ipv6-2-advpnlocalin] action pass
   [Hub1-security-policy-ipv6-2-advpnlocalin] quit
   ```

   # Configure a rule named **ipseclocalout** to allow Hub 1 to send IPsec negotiation packets to other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ipseclocalout
[Hub1-security-policy-ipv6-3-ipseclocalout] source-zone local
[Hub1-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Hub1-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::1
[Hub1-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::2 1::4
[Hub1-security-policy-ipv6-3-ipseclocalout] action pass
[Hub1-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 1 to receive IPsec negotiation packets from other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ipseclocalin
[Hub1-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Hub1-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Hub1-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::2 1::4
[Hub1-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::1
[Hub1-security-policy-ipv6-4-ipseclocalin] action pass
[Hub1-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 1 to send OSPF packets to other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ospflocalout
[Hub1-security-policy-ipv6-5-ospflocalout] source-zone local
[Hub1-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Hub1-security-policy-ipv6-5-ospflocalout] service ospf
[Hub1-security-policy-ipv6-5-ospflocalout] action pass
[Hub1-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 1 to receive OSPF packets from other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ospflocalin
[Hub1-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Hub1-security-policy-ipv6-6-ospflocalin] destination-zone local
[Hub1-security-policy-ipv6-6-ospflocalin] service ospf
[Hub1-security-policy-ipv6-6-ospflocalin] action pass
[Hub1-security-policy-ipv6-6-ospflocalin] quit
```

4. Configure the VAM client:

# Create VAM client **Hub1**.

```
<Hub1> system-view
[Hub1] vam client name Hub1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub1-vam-client-Hub1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub1-vam-client-Hub1] pre-shared-key simple 123456
```

# Set both the username and password to **hub1**.

```
[Hub1-vam-client-Hub1] user hub1 password simple hub1
```

# Specify the primary and secondary VAM servers.

```
[Hub1-vam-client-Hub1] server primary ipv6-address 1::11
[Hub1-vam-client-Hub1] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Hub1-vam-client-Hub1] client enable
[Hub1-vam-client-Hub1] quit
```

5. **Configure an IPsec profile:**

   # Configure IKE.

   ```
   [Hub1] ike keychain abc
   [Hub1-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
   [Hub1-ike-keychain-abc] quit
   [Hub1] ike profile abc
   [Hub1-ike-profile-abc] keychain abc
   [Hub1-ike-profile-abc] quit
   ```

   # Configure the IPsec profile.

   ```
   [Hub1] ipsec transform-set abc
   [Hub1-ipsec-transform-set-abc] encapsulation-mode transport
   [Hub1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
   [Hub1-ipsec-transform-set-abc] esp authentication-algorithm sha1
   [Hub1-ipsec-transform-set-abc] quit
   [Hub1] ipsec profile abc isakmp
   [Hub1-ipsec-profile-isakmp-abc] transform-set abc
   [Hub1-ipsec-profile-isakmp-abc] ike-profile abc
   [Hub1-ipsec-profile-isakmp-abc] quit
   ```

6. **Configure OSPFv3.**

   ```
   [Hub1] ospfv3 1
   [Hub1-ospfv3-1] router-id 0.0.0.1
   [Hub1-ospfv3-1] area 0
   [Hub1-ospfv3-1-area-0.0.0.0] quit
   [Hub1-ospfv3-1] quit
   ```

7. **Configure interface Tunnel 1.**

   ```
   [Hub1] interface tunnel1
   [Hub1-Tunnel1] vam ipv6 client Hub1
   [Hub1-Tunnel1] ospfv3 1 area 0
   [Hub1-Tunnel1] ospfv3 network-type broadcast
   [Hub1-Tunnel1] source gigabitethernet 1/0/1
   [Hub1-Tunnel1] tunnel protection ipsec profile abc
   [Hub1-Tunnel1] quit
   ```

Configuring Hub 2

1. **Assign IP addresses to interfaces:**

   # Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

   ```
   <Hub2> system-view
   [Hub2] interface gigabitethernet 1/0/1
   [Hub2-GigabitEthernet1/0/1] ipv6 address 1::2 64
   [Hub2-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

   ```
   [Hub2] interface tunnel 1 mode advpn gre ipv6
   [Hub2-Tunnel1] ipv6 address 192:168::2 64
   [Hub2-Tunnel1] ipv6 address fe80::2 link-local
   ```

2. **Add the interfaces to a security zone.**

   ```
   [Hub2] security-zone name untrust
   [Hub2-security-zone-Untrust] import interface gigabitethernet 1/0/1
   ```

```
[Hub2-security-zone-Untrust] import interface tunnel1
[Hub2-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Hub 2 to send VAM protocol packets to the VAM server.

```
[Hub2] security-policy ipv6
[Hub2-security-policy-ipv6] rule name advpnlocalout
[Hub2-security-policy-ipv6-1-advpnlocalout] source-zone local
[Hub2-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Hub2-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::2
[Hub2-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Hub2-security-policy-ipv6-1-ipseclocalout] action pass
[Hub2-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Hub 2 to receive VAM protocol packets from the VAM server.

```
[Hub2-security-policy-ipv6] rule name advpnlocalin
[Hub2-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Hub2-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Hub2-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Hub2-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::2
[Hub2-security-policy-ipv6-2-advpnlocalin] action pass
[Hub2-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Hub 2 to send IPsec negotiation packets to other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ipseclocalout
[Hub2-security-policy-ipv6-3-ipseclocalout] source-zone local
[Hub2-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Hub2-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::2
[Hub2-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::4
[Hub2-security-policy-ipv6-3-ipseclocalout] action pass
[Hub2-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 2 to receive IPsec negotiation packets from other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ipseclocalin
[Hub2-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Hub2-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Hub2-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::4
[Hub2-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::2
[Hub2-security-policy-ipv6-4-ipseclocalin] action pass
[Hub2-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 2 to send OSPF packets to other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ospflocalout
[Hub2-security-policy-ipv6-5-ospflocalout] source-zone local
[Hub2-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Hub2-security-policy-ipv6-5-ospflocalout] service ospf
[Hub2-security-policy-ipv6-5-ospflocalout] action pass
[Hub2-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 2 to receive OSPF packets from other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ospflocalin
[Hub2-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Hub2-security-policy-ipv6-6-ospflocalin] destination-zone local
[Hub2-security-policy-ipv6-6-ospflocalin] service ospf
[Hub2-security-policy-ipv6-6-ospflocalin] action pass
[Hub2-security-policy-ipv6-6-ospflocalin] quit
[Hub2-security-policy-ipv6] quit
```

4. Configure the VAM client:

# Create VAM client **Hub2**.

```
<Hub2> system-view
[Hub2] vam client name Hub2
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub2-vam-client-Hub2] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub2-vam-client-Hub2] pre-shared-key simple 123456
```

# Set both the username and password to **hub2**.

```
[Hub2-vam-client-Hub2] user hub2 password simple hub2
```

# Specify the primary and secondary VAM servers.

```
[Hub2-vam-client-Hub2] server primary ipv6-address 1::11
[Hub2-vam-client-Hub2] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Hub2-vam-client-Hub2] client enable
[Hub2-vam-client-Hub2] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Hub2] ike keychain abc
[Hub2-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Hub2-ike-keychain-abc] quit
[Hub2] ike profile abc
[Hub2-ike-profile-abc] keychain abc
[Hub2-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Hub2] ipsec transform-set abc
[Hub2-ipsec-transform-set-abc] encapsulation-mode transport
[Hub2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Hub2-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Hub2-ipsec-transform-set-abc] quit
[Hub2] ipsec profile abc isakmp
[Hub2-ipsec-profile-isakmp-abc] transform-set abc
[Hub2-ipsec-profile-isakmp-abc] ike-profile abc
[Hub2-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Hub2] ospfv3 1
[Hub2-ospfv3-1] router-id 0.0.0.2
[Hub2-ospfv3-1] area 0
```

```
[Hub2-ospfv3-1-area-0.0.0.0] quit
[Hub2-ospfv3-1] quit
```

7. Configure interface Tunnel 1.

```
[Hub2] interface tunnel1
[Hub2-Tunnel1] vam ipv6 client Hub2
[Hub2-Tunnel1] ospfv3 1 area 0
[Hub2-Tunnel1] ospfv3 network-type broadcast
[Hub2-Tunnel1] source gigabitethernet 1/0/1
[Hub2-Tunnel1] tunnel protection ipsec profile abc
[Hub2-Tunnel1] quit
```

Configuring Spoke 1

1. Assign IP addresses to interfaces:

   # Assign a global unicast address to interface GigabitEthernet 1/0/1.

```
<Spoke1> system-view
[Spoke1] interface gigabitethernet 1/0/1
[Spoke1-GigabitEthernet1/0/1] ipv6 address 1::3 64
[Spoke1-GigabitEthernet1/0/1] quit
```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Spoke1] interface tunnel 1 mode advpn gre ipv6
[Spoke1-Tunnel1] ipv6 address 192:168::3 64
[Spoke1-Tunnel1] ipv6 address fe80::3 link-local
```

   # Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke1] security-zone name untrust
[Spoke1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Spoke1-security-zone-Untrust] import interface tunnel1
[Spoke1-security-zone-Untrust] quit
[Spoke1] security-zone name trust
[Spoke1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke1-security-zone-Trust] quit
```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Spoke 1 to send VAM protocol packets to the VAM server.

```
[Spoke1] security-policy ipv6
[Spoke1-security-policy-ipv6] rule name advpnlocalout
[Spoke1-security-policy-ipv6-1-advpnlocalout] source-zone local
[Spoke1-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::3
[Spoke1-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Spoke1-security-policy-ipv6-1-ipseclocalout] action pass
[Spoke1-security-policy-ipv6-1-ipseclocalout] quit
```

   # Configure a rule named **advpnlocalin** to allow Spoke 1 to receive VAM protocol packets from the VAM server.

```
[Spoke1-security-policy-ipv6] rule name advpnlocalin
[Spoke1-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Spoke1-security-policy-ipv6-2-advpnlocalin] destination-zone local
```

```
[Spoke1-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Spoke1-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::3
[Spoke1-security-policy-ipv6-2-advpnlocalin] action pass
[Spoke1-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 1 to send IPsec negotiation packets to other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ipseclocalout
[Spoke1-security-policy-ipv6-3-ipseclocalout] source-zone local
[Spoke1-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::3
[Spoke1-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::4
[Spoke1-security-policy-ipv6-3-ipseclocalout] action pass
[Spoke1-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 1 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ipseclocalin
[Spoke1-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Spoke1-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Spoke1-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::4
[Spoke1-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::3
[Spoke1-security-policy-ipv6-4-ipseclocalin] action pass
[Spoke1-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 1 to send OSPF packets to other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ospflocalout
[Spoke1-security-policy-ipv6-5-ospflocalout] source-zone local
[Spoke1-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-5-ospflocalout] service ospf
[Spoke1-security-policy-ipv6-5-ospflocalout] action pass
[Spoke1-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 1 to receive OSPF packets from other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ospflocalin
[Spoke1-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Spoke1-security-policy-ipv6-6-ospflocalin] destination-zone local
[Spoke1-security-policy-ipv6-6-ospflocalin] service ospf
[Spoke1-security-policy-ipv6-6-ospflocalin] action pass
[Spoke1-security-policy-ipv6-6-ospflocalin] quit
[Spoke1-security-policy-ipv6] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke1**.

```
<Spoke1> system-view
[Spoke1] vam client name Spoke1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke1-vam-client-Spoke1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke1-vam-client-Spoke1] pre-shared-key simple 123456
```

# Set both the username and password to **spoke1**.

```
[Spoke1-vam-client-Spoke1] user spoke1 password simple spoke1
```

# Specify the primary and secondary VAM servers.

```
[Spoke1-vam-client-Spoke1] server primary ipv6-address 1::11
[Spoke1-vam-client-Spoke1] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Spoke1-vam-client-Spoke1] client enable
[Spoke1-vam-client-Spoke1] quit
```

5. Configure an IPsec profile:

   # Configure IKE.

```
[Spoke1] ike keychain abc
[Spoke1-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Spoke1-ike-keychain-abc] quit
[Spoke1] ike profile abc
[Spoke1-ike-profile-abc] keychain abc
[Spoke1-ike-profile-abc] quit
```

   # Configure the IPsec profile.

```
[Spoke1] ipsec transform-set abc
[Spoke1-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke1-ipsec-transform-set-abc] quit
[Spoke1] ipsec profile abc isakmp
[Spoke1-ipsec-profile-isakmp-abc] transform-set abc
[Spoke1-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Spoke1] ospfv3 1
[Spoke1-ospfv3-1] router-id 0.0.0.3
[Spoke1-ospfv3-1] area 0
[Spoke1-ospfv3-1-area-0.0.0.0] quit
[Spoke1-ospfv3-1] quit
```

7. Configure interface Tunnel 1. Set its DR priority to 0 to prevent Spoke 1 from participating in DR/BDR election.

```
[Spoke1] interface tunnel1 mode advpn gre ipv6
[Spoke1-Tunnel1] vam ipv6 client Spoke1
[Spoke1-Tunnel1] ospfv3 1 area 0
[Spoke1-Tunnel1] ospfv3 network-type broadcast
[Spoke1-Tunnel1] ospfv3 dr-priority 0
[Spoke1-Tunnel1] source gigabitethernet 1/0/1
[Spoke1-Tunnel1] tunnel protection ipsec profile abc
[Spoke1-Tunnel1] quit
```

Configuring Spoke 2

1. Assign IP addresses to interfaces:

   # Assign a global unicast address to interface GigabitEthernet 1/0/1.

```
<Spoke2> system-view
[Spoke2] interface gigabitethernet 1/0/1
[Spoke2-GigabitEthernet1/0/1] ipv6 address 1::4 64
```

```
[Spoke2-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Spoke2] interface tunnel 1 mode advpn gre ipv6
[Spoke2-Tunnel1] ipv6 address 192:168::4 64
[Spoke2-Tunnel1] ipv6 address fe80::4 link-local
```

# Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke2] security-zone name untrust
[Spoke2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Spoke2-security-zone-Untrust] import interface tunnel1
[Spoke2-security-zone-Untrust] quit
[Spoke2] security-zone name trust
[Spoke2-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke2-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Spoke 2 to send VAM protocol packets to the VAM server.

```
[Spoke2] security-policy ipv6
[Spoke2-security-policy-ipv6] rule name advpnlocalout
[Spoke2-security-policy-ipv6-1-advpnlocalout] source-zone local
[Spoke2-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::4
[Spoke2-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Spoke2-security-policy-ipv6-1-ipseclocalout] action pass
[Spoke2-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Spoke 2 to receive VAM protocol packets from the VAM server.

```
[Spoke2-security-policy-ipv6] rule name advpnlocalin
[Spoke2-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Spoke2-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Spoke2-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Spoke2-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::4
[Spoke2-security-policy-ipv6-2-advpnlocalin] action pass
[Spoke2-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 2 to send IPsec negotiation packets to other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ipseclocalout
[Spoke2-security-policy-ipv6-3-ipseclocalout] source-zone local
[Spoke2-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::4
[Spoke2-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::3
[Spoke2-security-policy-ipv6-3-ipseclocalout] action pass
[Spoke2-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 2 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ipseclocalin
[Spoke2-security-policy-ipv6-4-ipseclocalin] source-zone untrust
```

```
[Spoke2-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Spoke2-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::3
[Spoke2-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::4
[Spoke2-security-policy-ipv6-4-ipseclocalin] action pass
[Spoke2-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 2 to send OSPF packets to other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ospflocalout
[Spoke2-security-policy-ipv6-5-ospflocalout] source-zone local
[Spoke2-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-5-ospflocalout] service ospf
[Spoke2-security-policy-ipv6-5-ospflocalout] action pass
[Spoke2-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 2 to receive OSPF packets from other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ospflocalin
[Spoke2-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Spoke2-security-policy-ipv6-6-ospflocalin] destination-zone local
[Spoke2-security-policy-ipv6-6-ospflocalin] service ospf
[Spoke2-security-policy-ipv6-6-ospflocalin] action pass
[Spoke2-security-policy-ipv6-6-ospflocalin] quit
[Spoke2-security-policy-ipv6] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke2**.

```
<Spoke2> system-view
[Spoke2] vam client name Spoke2
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke2-vam-client-Spoke2] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke2-vam-client-Spoke2] pre-shared-key simple 123456
```

# Set both the username and password to **spoke2**.

```
[Spoke2-vam-client-Spoke2] user spoke2 password simple spoke2
```

# Specify the primary and secondary VAM servers.

```
[Spoke2-vam-client-Spoke2] server primary ipv6-address 1::11
[Spoke2-vam-client-Spoke2] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Spoke2-vam-client-Spoke2] client enable
[Spoke2-vam-client-Spoke2] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Spoke2] ike keychain abc
[Spoke2-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Spoke2-ike-keychain-abc] quit
[Spoke2] ike profile abc
[Spoke2-ike-profile-abc] keychain abc
[Spoke2-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Spoke2] ipsec transform-set abc
```

```
[Spoke2-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke2-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke2-ipsec-transform-set-abc] quit
[Spoke2] ipsec profile abc isakmp
[Spoke2-ipsec-profile-isakmp-abc] transform-set abc
[Spoke2-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke2-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Spoke2] ospfv3 1
[Spoke2-ospfv3-1] router-id 0.0.0.4
[Spoke2-ospfv3-1] area 0
[Spoke2-ospfv3-1-area-0.0.0.0] quit
[Spoke2-ospfv3-1] quit
```

7. Configure interface Tunnel 1. Set its DR priority to 0 to prevent Spoke 2 from participating in DR/BDR election.

```
[Spoke2] interface tunnel1 mode advpn gre ipv6
[Spoke2-Tunnel1] vam ipv6 client Spoke2
[Spoke2-Tunnel1] ospfv3 1 area 0
[Spoke2-Tunnel1] ospfv3 network-type broadcast
[Spoke2-Tunnel1] ospfv3 dr-priority 0
[Spoke2-Tunnel1] source gigabitethernet 1/0/1
[Spoke2-Tunnel1] tunnel protection ipsec profile abc
[Spoke2-Tunnel1] quit
```

Verifying the configuration

# Display IPv6 address mapping information for all VAM clients registered with the primary VAM server.

```
[PrimaryServer] display vam server ipv6 address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address       Public address          Type   NAT  Holding time
0          192:168::1            1::1                    Hub    No   0H 52M  7S
0          192:168::2            1::2                    Hub    No   0H 47M 31S
0          192:168::3            1::3                    Spoke  No   0H 28M 25S
0          192:168::4            1::4                    Spoke  No   0H 19M 15S
```

# Display IPv6 address mapping information for all VAM clients registered with the secondary VAM server.

```
[SecondaryServer] display vam server ipv6 address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address       Public address          Type   NAT  Holding time
0          192:168::1            1::1                    Hub    No   0H 52M  7S
0          192:168::2            1::2                    Hub    No   0H 47M 31S
0          192:168::3            1::3                    Spoke  No   0H 28M 25S
0          192:168::4            1::4                    Spoke  No   0H 19M 15S
```

The output shows that Hub 1, Hub 2, Spoke 1, and Spoke 2 have all registered their address mapping information with the VAM servers.

# Display IPv6 ADVPN tunnel information on Hubs. This example uses Hub 1.

```
[Hub1] display advpn ipv6 session
Interface         : Tunnel1
Number of sessions: 3
Private address       Public address        Port  Type  State      Holding time
192:168::2            1::2                  --    H-H   Success    0H 46M  8S
192:168::3            1::3                  --    H-S   Success    0H 27M 27S
192:168::4            1::4                  --    H-S   Success    0H 18M 18S
```

The output shows that Hub 1 has established a permanent tunnel to Hub 2, Spoke 1, and Spoke 2.

# Display IPv6 ADVPN tunnel information on Spoke 1.

```
[Spoke1] display advpn ipv6 session
Interface         : Tunnel1
Number of sessions: 2
Private address       Public address        Port  Type  State      Holding time
192:168::1            1::1                  --    S-H   Success    0H 46M  8S
192:168::2            1::2                  --    S-H   Success    0H 46M  8S
```

The output shows that Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.

# Verify that Spoke 1 can ping the private address 192:168::4 of Spoke 2.

```
[Spoke1] ping ipv6 192:168::4
Ping6(56 data bytes) 192:168::4 --> 192:168::4, press CTRL_C to break
56 bytes from 192:168::4, icmp_seq=0 hlim=64 time=3.000 ms
56 bytes from 192:168::4, icmp_seq=1 hlim=64 time=0.000 ms
56 bytes from 192:168::4, icmp_seq=2 hlim=64 time=1.000 ms
56 bytes from 192:168::4, icmp_seq=3 hlim=64 time=1.000 ms
56 bytes from 192:168::4, icmp_seq=4 hlim=64 time=1.000 ms

--- Ping6 statistics for 192:168::4 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.200/3.000/0.980 ms
```

# Display IPv6 ADVPN tunnel information on Spokes. This example uses Spoke 1.

```
[Spoke1] display advpn ipv6 session
Interface         : Tunnel1
Number of sessions: 3
Private address       Public address        Port  Type  State      Holding time
192:168::1            1::1                  --    S-H   Success    0H 46M  8S
192:168::2            1::2                  --    S-H   Success    0H 46M  8S
192.168::4            1::4                  --    S-S   Success    0H  0M  1S
```

The output shows the following information:

- Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.
- Spoke 1 has established a temporary spoke-spoke tunnel to Spoke 2.

Example: Configuring IPv4 hub-spoke ADVPN

Network configuration

As shown in Figure 9, the primary and secondary VAM servers manage and maintain VAM client information for all hubs and spokes. The AAA server performs authentication and accounting for VAM clients. The two hubs back up each other, and perform data forwarding and route exchange.

Establish a permanent ADVPN tunnel between each spoke and each hub.

Figure 9 Network diagram

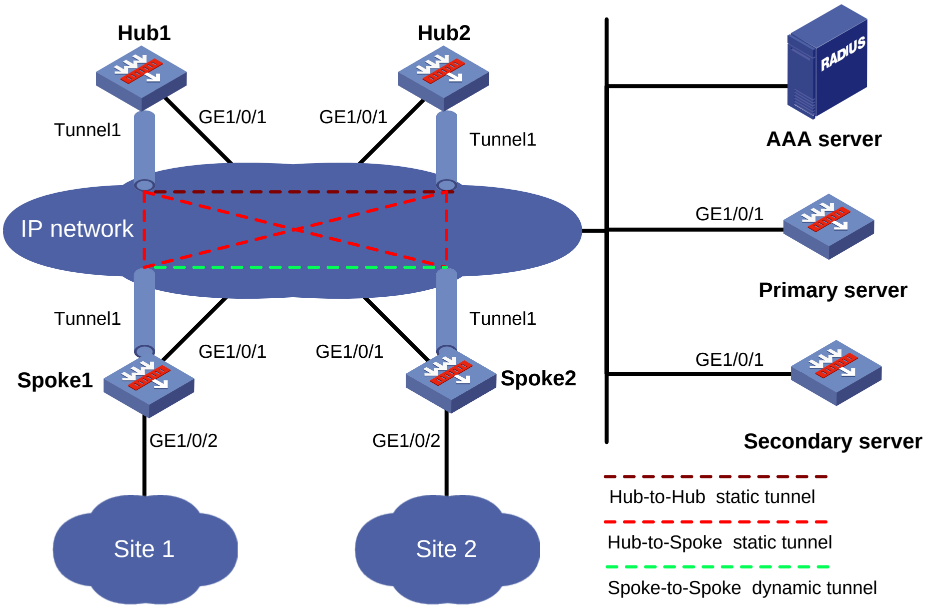


Table 3 Interface and IP address assignment

Device	Interface	IP address	Device	Interface	IP address
Hub 1	GE1/0/1	1.0.0.1/24	Spoke 1	GE1/0/1	1.0.0.3/24
	Tunnel1	192.168.0.1/24		GE1/0/2	192.168.1.1/24
Hub 2	GE1/0/1	1.0.0.2/24		Tunnel1	192.168.0.3/24
	Tunnel1	192.168.0.2/24	Spoke 2	GE1/0/1	1.0.0.4/24
AAA server		1.0.0.10/24		GE1/0/2	192.168.2.1/24
Primary server	GE1/0/1	1.0.0.11/24		Tunnel1	192.168.0.4/24
Secondary server	GE1/0/1	1.0.0.12/24			

Configuring the primary VAM server

1. Assign an IP address to interface GigabitEthernet 1/0/1.

```
<PrimaryServer> system-view
[PrimaryServer] interface gigabitethernet 1/0/1
[PrimaryServer-GigabitEthernet1/0/1] ip address 1.0.0.11 255.255.255.0
[PrimaryServer-GigabitEthernet1/0/1] quit
```

2. Add the interface to a security zone.

```
[PrimaryServer] security-zone name untrust
[PrimaryServer-security-zone-Untrust] import interface gigabitethernet 1/0/1
[PrimaryServer-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow the VAM server to send VAM protocol packets to the VAM clients.

```
[PrimaryServer] security-policy ip
[PrimaryServer-security-policy-ip] rule name advpnlocalout
```

```
[PrimaryServer-security-policy-ip-1-advpnlocalout] source-zone local
[PrimaryServer-security-policy-ip-1-advpnlocalout] destination-zone untrust
[PrimaryServer-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.11
[PrimaryServer-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.1
1.0.0.12
[PrimaryServer-security-policy-ip-1-ipseclocalout] action pass
[PrimaryServer-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow the VAM server to receive VAM protocol packets from the VAM clients.

```
[PrimaryServer-security-policy-ip] rule name advpnlocalin
[PrimaryServer-security-policy-ip-2-advpnlocalin] source-zone untrust
[PrimaryServer-security-policy-ip-2-advpnlocalin] destination-zone local
[PrimaryServer-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.1 1.0.0.12
[PrimaryServer-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.11
[PrimaryServer-security-policy-ip-2-advpnlocalin] action pass
[PrimaryServer-security-policy-ip-2-advpnlocalin] quit
[PrimaryServer-security-policy-ip] quit
```

4. Configure AAA:

   # Configure RADIUS scheme **abc**.

```
<PrimaryServer> system-view
[PrimaryServer] radius scheme abc
[PrimaryServer-radius-abc] primary authentication 1.0.0.10 1812
[PrimaryServer-radius-abc] primary accounting 1.0.0.10 1813
[PrimaryServer-radius-abc] key authentication simple 123
[PrimaryServer-radius-abc] key accounting simple 123
[PrimaryServer-radius-abc] user-name-format without-domain
[PrimaryServer-radius-abc] quit
[PrimaryServer] radius session-control enable
```

   # Configure AAA methods for ISP domain **abc**.

```
[PrimaryServer] domain abc
[PrimaryServer-isp-abc] authentication advpn radius-scheme abc
[PrimaryServer-isp-abc] accounting advpn radius-scheme abc
[PrimaryServer-isp-abc] quit
[PrimaryServer] domain default enable abc
```

5. Configure the VAM server:

   # Create ADVPN domain **abc**.

```
[PrimaryServer] vam server advpn-domain abc id 1
```

   # Create hub group 0.

```
[PrimaryServer-vam-server-domain-abc] hub-group 0
```

   # Specify hub private IPv4 addresses.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub private-address 192.168.0.1
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub private-address 192.168.0.2
```

   # Specify a spoke private IPv4 network.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] spoke private-address network
192.168.0.0 255.255.255.0
[PrimaryServer-vam-server-domain-abc-hub-group-0] quit
```

   # Set the preshared key to **123456**.

```
[PrimaryServer-vam-server-domain-abc] pre-shared-key simple 123456
```

# Set the authentication mode to CHAP.

```
[PrimaryServer-vam-server-domain-abc] authentication-method chap
```

# Enable the VAM server for the ADVPN domain.

```
[PrimaryServer-vam-server-domain-abc] server enable
[PrimaryServer-vam-server-domain-abc] quit
```

Configuring the secondary VAM server

# Configure the secondary VAM server in the same way that the primary server is configured. (Details not shown.)

Configuring Hub 1

1. Assign IP addresses to interfaces:

   # Assign an IPv4 address to GigabitEthernet 1/0/1.

   ```
   <Hub1> system-view
   [Hub1] interface gigabitethernet 1/0/1
   [Hub1-GigabitEthernet1/0/1] ip address 1.0.0.1 255.255.255.0
   [Hub1-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

   ```
   [Hub1] interface tunnel1 mode advpn gre
   [Hub1-Tunnel1] ip address 192.168.0.1 255.255.255.0
   ```

2. Add the interfaces to a security zone:

   ```
   [Hub1] security-zone name untrust
   [Hub1-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Hub1-security-zone-Untrust] import interface tunnel1
   [Hub1-security-zone-Untrust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Hub 1 to send VAM protocol packets to the VAM server.

   ```
   [Hub1] security-policy ip
   [Hub1-security-policy-ip] rule name advpnlocalout
   [Hub1-security-policy-ip-1-advpnlocalout] source-zone local
   [Hub1-security-policy-ip-1-advpnlocalout] destination-zone untrust
   [Hub1-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.1
   [Hub1-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
   [Hub1-security-policy-ip-1-ipseclocalout] action pass
   [Hub1-security-policy-ip-1-ipseclocalout] quit
   ```

   # Configure a rule named **advpnlocalin** to allow Hub 1 to receive VAM protocol packets from the VAM server.

   ```
   [Hub1-security-policy-ip] rule name advpnlocalin
   [Hub1-security-policy-ip-2-advpnlocalin] source-zone untrust
   [Hub1-security-policy-ip-2-advpnlocalin] destination-zone local
   [Hub1-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
   [Hub1-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.1
   [Hub1-security-policy-ip-2-advpnlocalin] action pass
   [Hub1-security-policy-ip-2-advpnlocalin] quit
   ```

   # Configure a rule named **ipseclocalout** to allow Hub 1 to send IPsec negotiation packets to other VAM clients.

   ```
   [Hub1-security-policy-ip] rule name ipseclocalout
   ```

```
[Hub1-security-policy-ip-3-ipseclocalout] source-zone local
[Hub1-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Hub1-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.1
[Hub1-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.2 1.0.0.4
[Hub1-security-policy-ip-3-ipseclocalout] action pass
[Hub1-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 1 to receive IPsec negotiation packets from other VAM clients.

```
[Hub1-security-policy-ip] rule name ipseclocalin
[Hub1-security-policy-ip-4-ipseclocalin] source-zone untrust
[Hub1-security-policy-ip-4-ipseclocalin] destination-zone local
[Hub1-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.2 1.0.0.4
[Hub1-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.1
[Hub1-security-policy-ip-4-ipseclocalin] action pass
[Hub1-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 1 to send OSPF packets to other VAM clients.

```
[Hub1-security-policy-ip] rule name ospflocalout
[Hub1-security-policy-ip-5-ospflocalout] source-zone local
[Hub1-security-policy-ip-5-ospflocalout] destination-zone untrust
[Hub1-security-policy-ip-5-ospflocalout] service ospf
[Hub1-security-policy-ip-5-ospflocalout] action pass
[Hub1-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 1 to receive OSPF packets from other VAM clients.

```
[Hub1-security-policy-ip] rule name ospflocalin
[Hub1-security-policy-ip-6-ospflocalin] source-zone untrust
[Hub1-security-policy-ip-6-ospflocalin] destination-zone local
[Hub1-security-policy-ip-6-ospflocalin] service ospf
[Hub1-security-policy-ip-6-ospflocalin] action pass
[Hub1-security-policy-ip-6-ospflocalin] quit
```

4. Configure the VAM client:

# Create VAM client **Hub1**.

```
<Hub1> system-view
[Hub1] vam client name Hub1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub1-vam-client-Hub1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub1-vam-client-Hub1] pre-shared-key simple 123456
```

# Set both the username and password to **hub1**.

```
[Hub1-vam-client-Hub1] user hub1 password simple hub1
```

# Specify the primary and secondary VAM servers.

```
[Hub1-vam-client-Hub1] server primary ip-address 1.0.0.11
[Hub1-vam-client-Hub1] server secondary ip-address 1.0.0.12
```

# Enable the VAM client.

```
[Hub1-vam-client-Hub1] client enable
[Hub1-vam-client-Hub1] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Hub1] ike keychain abc
[Hub1-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
[Hub1-ike-keychain-abc] quit
[Hub1] ike profile abc
[Hub1-ike-profile-abc] keychain abc
[Hub1-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Hub1] ipsec transform-set abc
[Hub1-ipsec-transform-set-abc] encapsulation-mode transport
[Hub1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Hub1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Hub1-ipsec-transform-set-abc] quit
[Hub1] ipsec profile abc isakmp
[Hub1-ipsec-profile-isakmp-abc] transform-set abc
[Hub1-ipsec-profile-isakmp-abc] ike-profile abc
[Hub1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise the private network.

```
[Hub1] ospf 1
[Hub1-ospf-1] area 0
[Hub1-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[Hub1-ospf-1-area-0.0.0.0] quit
[Hub1-ospf-1] quit
```

7. Configure interface Tunnel 1.

```
[Hub1] interface tunnel1
[Hub1-Tunnel1] vam client Hub1
[Hub1-Tunnel1] ospf network-type p2mp
[Hub1-Tunnel1] source gigabitethernet 1/0/1
[Hub1-Tunnel1] tunnel protection ipsec profile abc
[Hub1-Tunnel1] quit
```

Configuring Hub 2

1. Assign IP addresses to interfaces:

# Assign an IPv4 address to GigabitEthernet 1/0/1.

```
<Hub2> system-view
[Hub2] interface gigabitethernet 1/0/1
[Hub2-GigabitEthernet1/0/1] ip address 1.0.0.2 255.255.255.0
[Hub2-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

```
[Hub2] interface tunnel1 mode advpn gre
[Hub2-Tunnel1] ip address 192.168.0.2 255.255.255.0
```

2. Add the interfaces to a security zone.

```
[Hub2] security-zone name untrust
[Hub2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Hub2-security-zone-Untrust] import interface tunnel1
[Hub2-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Hub 2 to send VAM protocol packets to the VAM server.

```
[Hub2] security-policy ip
[Hub2-security-policy-ip] rule name advpnlocalout
[Hub2-security-policy-ip-1-advpnlocalout] source-zone local
[Hub2-security-policy-ip-1-advpnlocalout] destination-zone untrust
[Hub2-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.2
[Hub2-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
[Hub2-security-policy-ip-1-ipseclocalout] action pass
[Hub2-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Hub 2 to receive VAM protocol packets from the VAM server.

```
[Hub2-security-policy-ip] rule name advpnlocalin
[Hub2-security-policy-ip-2-advpnlocalin] source-zone untrust
[Hub2-security-policy-ip-2-advpnlocalin] destination-zone local
[Hub2-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
[Hub2-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.2
[Hub2-security-policy-ip-2-advpnlocalin] action pass
[Hub2-security-policy-ip-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Hub 2 to send IPsec negotiation packets to other VAM clients.

```
[Hub2-security-policy-ip] rule name ipseclocalout
[Hub2-security-policy-ip-3-ipseclocalout] source-zone local
[Hub2-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Hub2-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.2
[Hub2-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.4
[Hub2-security-policy-ip-3-ipseclocalout] action pass
[Hub2-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 2 to receive IPsec negotiation packets from other VAM clients.

```
[Hub2-security-policy-ip] rule name ipseclocalin
[Hub2-security-policy-ip-4-ipseclocalin] source-zone untrust
[Hub2-security-policy-ip-4-ipseclocalin] destination-zone local
[Hub2-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.4
[Hub2-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.2
[Hub2-security-policy-ip-4-ipseclocalin] action pass
[Hub2-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 2 to send OSPF packets to other VAM clients.

```
[Hub2-security-policy-ip] rule name ospflocalout
[Hub2-security-policy-ip-5-ospflocalout] source-zone local
[Hub2-security-policy-ip-5-ospflocalout] destination-zone untrust
[Hub2-security-policy-ip-5-ospflocalout] service ospf
[Hub2-security-policy-ip-5-ospflocalout] action pass
[Hub2-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 2 to receive OSPF packets from other VAM clients.

```
[Hub2-security-policy-ip] rule name ospflocalin
[Hub2-security-policy-ip-6-ospflocalin] source-zone untrust
```

```
[Hub2-security-policy-ip-6-ospflocalin] destination-zone local
[Hub2-security-policy-ip-6-ospflocalin] service ospf
[Hub2-security-policy-ip-6-ospflocalin] action pass
[Hub2-security-policy-ip-6-ospflocalin] quit
[Hub2-security-policy-ip] quit
```

4. Configure the VAM client:

   # Create VAM client **Hub2**.

   ```
   <Hub2> system-view
   [Hub2] vam client name Hub2
   ```

   # Specify ADVPN domain **abc** for the VAM client.

   ```
   [Hub2-vam-client-Hub2] advpn-domain abc
   ```

   # Set the preshared key to **123456**.

   ```
   [Hub2-vam-client-Hub2] pre-shared-key simple 123456
   ```

   # Set both the username and password to **hub2**.

   ```
   [Hub2-vam-client-Hub2] user hub2 password simple hub2
   ```

   # Specify the primary and secondary VAM servers.

   ```
   [Hub2-vam-client-Hub2] server primary ip-address 1.0.0.11
   [Hub2-vam-client-Hub2] server secondary ip-address 1.0.0.12
   ```

   # Enable the VAM client.

   ```
   [Hub2-vam-client-Hub2] client enable
   [Hub2-vam-client-Hub2] quit
   ```

5. Configure an IPsec profile:

   # Configure IKE.

   ```
   [Hub2] ike keychain abc
   [Hub2-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
   [Hub2-ike-keychain-abc] quit
   [Hub2] ike profile abc
   [Hub2-ike-profile-abc] keychain abc
   [Hub2-ike-profile-abc] quit
   ```

   # Configure the IPsec profile.

   ```
   [Hub2] ipsec transform-set abc
   [Hub2-ipsec-transform-set-abc] encapsulation-mode transport
   [Hub2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
   [Hub2-ipsec-transform-set-abc] esp authentication-algorithm sha1
   [Hub2-ipsec-transform-set-abc] quit
   [Hub2] ipsec profile abc isakmp
   [Hub2-ipsec-profile-isakmp-abc] transform-set abc
   [Hub2-ipsec-profile-isakmp-abc] ike-profile abc
   [Hub2-ipsec-profile-isakmp-abc] quit
   ```

6. Configure OSPF to advertise the private network.

   ```
   [Hub2] ospf 1
   [Hub2-ospf-1] area 0
   [Hub2-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
   [Hub2-ospf-1-area-0.0.0.0] quit
   [Hub2-ospf-1] quit
   ```

7. Configure interface Tunnel 1.

   ```
   [Hub2] interface tunnel1
   ```

```
[Hub2-Tunnel1] vam client Hub2
[Hub2-Tunnel1] ospf network-type p2mp
[Hub2-Tunnel1] source gigabitethernet 1/0/1
[Hub2-Tunnel1] tunnel protection ipsec profile abc
[Hub2-Tunnel1] quit
```

Configuring Spoke 1

1. Assign IP addresses to interfaces:

   # Assign an IPv4 address to interface GigabitEthernet 1/0/1.

   ```
   <Spoke1> system-view
   [Spoke1] interface gigabitethernet 1/0/1
   [Spoke1-GigabitEthernet1/0/1] ip address 1.0.0.3 255.255.255.0
   [Spoke1-GigabitEthernet1/0/1] quit
   ```

   # Create interface Tunnel 1 and set its tunnel mode to GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

   ```
   [Spoke1] interface tunnel 1 mode advpn gre
   [Spoke1-Tunnel1] ip address 192.168.0.3 255.255.255.0
   ```

   # Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

   ```
   [Spoke1] security-zone name untrust
   [Spoke1-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Spoke1-security-zone-Untrust] import interface tunnel1
   [Spoke1-security-zone-Untrust] quit
   [Spoke1] security-zone name trust
   [Spoke1-security-zone-Trust] import interface gigabitethernet 1/0/2
   [Spoke1-security-zone-Trust] quit
   ```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Spoke 1 to send VAM protocol packets to the VAM server.

   ```
   [Spoke1] security-policy ip
   [Spoke1-security-policy-ip] rule name advpnlocalout
   [Spoke1-security-policy-ip-1-advpnlocalout] source-zone local
   [Spoke1-security-policy-ip-1-advpnlocalout] destination-zone untrust
   [Spoke1-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.3
   [Spoke1-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
   [Spoke1-security-policy-ip-1-ipseclocalout] action pass
   [Spoke1-security-policy-ip-1-ipseclocalout] quit
   ```

   # Configure a rule named **advpnlocalin** to allow Spoke 1 to receive VAM protocol packets from the VAM server.

   ```
   [Spoke1-security-policy-ip] rule name advpnlocalin
   [Spoke1-security-policy-ip-2-advpnlocalin] source-zone untrust
   [Spoke1-security-policy-ip-2-advpnlocalin] destination-zone local
   [Spoke1-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
   [Spoke1-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.3
   [Spoke1-security-policy-ip-2-advpnlocalin] action pass
   [Spoke1-security-policy-ip-2-advpnlocalin] quit
   ```

   # Configure a rule named **ipseclocalout** to allow Spoke 1 to send IPsec negotiation packets to other VAM clients.

   ```
   [Spoke1-security-policy-ip] rule name ipseclocalout
   ```

```
[Spoke1-security-policy-ip-3-ipseclocalout] source-zone local
[Spoke1-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Spoke1-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.3
[Spoke1-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.4
[Spoke1-security-policy-ip-3-ipseclocalout] action pass
[Spoke1-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 1 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke1-security-policy-ip] rule name ipseclocalin
[Spoke1-security-policy-ip-4-ipseclocalin] source-zone untrust
[Spoke1-security-policy-ip-4-ipseclocalin] destination-zone local
[Spoke1-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.4
[Spoke1-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.3
[Spoke1-security-policy-ip-4-ipseclocalin] action pass
[Spoke1-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 1 to send OSPF packets to other VAM clients.

```
[Spoke1-security-policy-ip] rule name ospflocalout
[Spoke1-security-policy-ip-5-ospflocalout] source-zone local
[Spoke1-security-policy-ip-5-ospflocalout] destination-zone untrust
[Spoke1-security-policy-ip-5-ospflocalout] service ospf
[Spoke1-security-policy-ip-5-ospflocalout] action pass
[Spoke1-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 1 to receive OSPF packets from other VAM clients.

```
[Spoke1-security-policy-ip] rule name ospflocalin
[Spoke1-security-policy-ip-6-ospflocalin] source-zone untrust
[Spoke1-security-policy-ip-6-ospflocalin] destination-zone local
[Spoke1-security-policy-ip-6-ospflocalin] service ospf
[Spoke1-security-policy-ip-6-ospflocalin] action pass
[Spoke1-security-policy-ip-6-ospflocalin] quit
[Spoke1-security-policy-ip] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke1**.

```
<Spoke1> system-view
[Spoke1] vam client name Spoke1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke1-vam-client-Spoke1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke1-vam-client-Spoke1] pre-shared-key simple 123456
```

# Set both the username and password to **spoke1**.

```
[Spoke1-vam-client-Spoke1] user spoke1 password simple spoke1
```

# Specify the primary and secondary VAM servers.

```
[Spoke1-vam-client-Spoke1] server primary ip-address 1.0.0.11
[Spoke1-vam-client-Spoke1] server secondary ip-address 1.0.0.12
```

# Enable the VAM client.

```
[Spoke1-vam-client-Spoke1] client enable
[Spoke1-vam-client-Spoke1] quit
```

5. Configure an IPsec profile:

   # Configure IKE.

```
[Spoke1] ike keychain abc
[Spoke1-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
[Spoke1-ike-keychain-abc] quit
[Spoke1] ike profile abc
[Spoke1-ike-profile-abc] keychain abc
[Spoke1-ike-profile-abc] quit
```

   # Configure the IPsec profile.

```
[Spoke1] ipsec transform-set abc
[Spoke1-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke1-ipsec-transform-set-abc] quit
[Spoke1] ipsec profile abc isakmp
[Spoke1-ipsec-profile-isakmp-abc] transform-set abc
[Spoke1-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise private networks.

```
[Spoke1] ospf 1
[Spoke1-ospf-1] area 0
[Spoke1-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[Spoke1-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[Spoke1-ospf-1-area-0.0.0.0] quit
[Spoke1-ospf-1] quit
```

7. Configure interface Tunnel 1.

```
[Spoke1] interface tunnel1
[Spoke1-Tunnel1] vam client Spoke1
[Spoke1-Tunnel1] ospf network-type p2mp
[Spoke1-Tunnel1] source gigabitethernet 1/0/1
[Spoke1-Tunnel1] tunnel protection ipsec profile abc
[Spoke1-Tunnel1] quit
```

Configuring Spoke 2

1. Assign IP addresses to interfaces:

   # Assign an IPv4 address to interface GigabitEthernet 1/0/1.

```
<Spoke2> system-view
[Spoke2] interface gigabitethernet 1/0/1
[Spoke2-GigabitEthernet1/0/1] ip address 1.0.0.4 255.255.255.0
[Spoke2-GigabitEthernet1/0/1] quit
```

   # Create interface Tunnel 1 and set its tunnel mode to GRE-encapsulated IPv4 ADVPN tunnel mode. Then, assign an IPv4 address to the interface.

```
[Spoke2] interface tunnel 1 mode advpn gre
[Spoke2-Tunnel1] ip address 192.168.0.4 255.255.255.0
```

   # Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke2] security-zone name untrust
[Spoke2-security-zone-Untrust] import interface gigabitethernet 1/0/1
```

```
[Spoke2-security-zone-Untrust] import interface tunnel1
[Spoke2-security-zone-Untrust] quit
[Spoke2] security-zone name trust
[Spoke2-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke2-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Spoke 2 to send VAM protocol packets to the VAM server.

```
[Spoke2] security-policy ip
[Spoke2-security-policy-ip] rule name advpnlocalout
[Spoke2-security-policy-ip-1-advpnlocalout] source-zone local
[Spoke2-security-policy-ip-1-advpnlocalout] destination-zone untrust
[Spoke2-security-policy-ip-1-advpnlocalout] source-ip-host 1.0.0.4
[Spoke2-security-policy-ip-1-advpnlocalout] destination-ip-range 1.0.0.10 1.0.0.12
[Spoke2-security-policy-ip-1-ipseclocalout] action pass
[Spoke2-security-policy-ip-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Spoke 2 to receive VAM protocol packets from the VAM server.

```
[Spoke2-security-policy-ip] rule name advpnlocalin
[Spoke2-security-policy-ip-2-advpnlocalin] source-zone untrust
[Spoke2-security-policy-ip-2-advpnlocalin] destination-zone local
[Spoke2-security-policy-ip-2-advpnlocalin] source-ip-range 1.0.0.10 1.0.0.12
[Spoke2-security-policy-ip-2-advpnlocalin] destination-ip-host 1.0.0.4
[Spoke2-security-policy-ip-2-advpnlocalin] action pass
[Spoke2-security-policy-ip-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 2 to send IPsec negotiation packets to other VAM clients.

```
[Spoke2-security-policy-ip] rule name ipseclocalout
[Spoke2-security-policy-ip-3-ipseclocalout] source-zone local
[Spoke2-security-policy-ip-3-ipseclocalout] destination-zone untrust
[Spoke2-security-policy-ip-3-ipseclocalout] source-ip-host 1.0.0.4
[Spoke2-security-policy-ip-3-ipseclocalout] destination-ip-range 1.0.0.1 1.0.0.3
[Spoke2-security-policy-ip-3-ipseclocalout] action pass
[Spoke2-security-policy-ip-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 2 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke2-security-policy-ip] rule name ipseclocalin
[Spoke2-security-policy-ip-4-ipseclocalin] source-zone untrust
[Spoke2-security-policy-ip-4-ipseclocalin] destination-zone local
[Spoke2-security-policy-ip-4-ipseclocalin] source-ip-range 1.0.0.1 1.0.0.3
[Spoke2-security-policy-ip-4-ipseclocalin] destination-ip-host 1.0.0.4
[Spoke2-security-policy-ip-4-ipseclocalin] action pass
[Spoke2-security-policy-ip-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 2 to send OSPF packets to other VAM clients.

```
[Spoke2-security-policy-ip] rule name ospflocalout
[Spoke2-security-policy-ip-5-ospflocalout] source-zone local
[Spoke2-security-policy-ip-5-ospflocalout] destination-zone untrust
[Spoke2-security-policy-ip-5-ospflocalout] service ospf
```

```
[Spoke2-security-policy-ip-5-ospflocalout] action pass
[Spoke2-security-policy-ip-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 2 to receive OSPF packets from other VAM clients.

```
[Spoke2-security-policy-ip] rule name ospflocalin
[Spoke2-security-policy-ip-6-ospflocalin] source-zone untrust
[Spoke2-security-policy-ip-6-ospflocalin] destination-zone local
[Spoke2-security-policy-ip-6-ospflocalin] service ospf
[Spoke2-security-policy-ip-6-ospflocalin] action pass
[Spoke2-security-policy-ip-6-ospflocalin] quit
[Spoke2-security-policy-ip] quit
```

4. Configure the VAM client:

   # Create VAM client **Spoke2**.

```
<Spoke2> system-view
[Spoke2] vam client name Spoke2
```

   # Specify ADVPN domain **abc** for the VAM client.

```
[Spoke2-vam-client-Spoke2] advpn-domain abc
```

   # Set the preshared key to **123456**.

```
[Spoke2-vam-client-Spoke2] pre-shared-key simple 123456
```

   # Set both the username and password to **spoke2**.

```
[Spoke2-vam-client-Spoke2] user spoke2 password simple spoke2
```

   # Specify the primary and secondary VAM servers.

```
[Spoke2-vam-client-Spoke2] server primary ip-address 1.0.0.11
[Spoke2-vam-client-Spoke2] server secondary ip-address 1.0.0.12
```

   # Enable the VAM client.

```
[Spoke2-vam-client-Spoke2] client enable
[Spoke2-vam-client-Spoke2] quit
```

5. Configure an IPsec profile:

   # Configure IKE.

```
[Spoke2] ike keychain abc
[Spoke2-ike-keychain-abc] pre-shared-key address 0.0.0.0 0.0.0.0 key simple 123456
[Spoke2-ike-keychain-abc] quit
[Spoke2] ike profile abc
[Spoke2-ike-profile-abc] keychain abc
[Spoke2-ike-profile-abc] quit
```

   # Configure the IPsec profile.

```
[Spoke2] ipsec transform-set abc
[Spoke2-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke2-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke2-ipsec-transform-set-abc] quit
[Spoke2] ipsec profile abc isakmp
[Spoke2-ipsec-profile-isakmp-abc] transform-set abc
[Spoke2-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke2-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPF to advertise private networks.

```
[Spoke2] ospf 1
```

```
[Spoke2-ospf-1] area 0
[Spoke2-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[Spoke2-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[Spoke2-ospf-1-area-0.0.0.0] quit
[Spoke2-ospf-1] quit
```

7. Configure interface Tunnel 1.

```
[Spoke2] interface tunnel1
[Spoke2-Tunnel1] vam client Spoke2
[Spoke2-Tunnel1] ospf network-type p2mp
[Spoke2-Tunnel1] source gigabitethernet 1/0/1
[Spoke2-Tunnel1] tunnel protection ipsec profile abc
[Spoke2-Tunnel1] quit
```

Verifying the configuration

# Display IPv4 address mapping information for all VAM clients registered with the primary VAM server.

```
[PrimaryServer] display vam server address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address  Public address              Type   NAT  Holding time
0          192.168.0.1      1.0.0.1                     Hub    No   0H 52M  7S
0          192.168.0.2      1.0.0.2                     Hub    No   0H 47M 31S
0          192.168.0.3      1.0.0.3                     Spoke  No   0H 28M 25S
0          192.168.0.4      1.0.0.4                     Spoke  No   0H 19M 15S
```

# Display IPv4 address mapping information for all VAM clients registered with the secondary VAM server.

```
[SecondaryServer] display vam server address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address  Public address              Type   NAT  Holding time
0          192.168.0.1      1.0.0.1                     Hub    No   0H 52M  7S
0          192.168.0.2      1.0.0.2                     Hub    No   0H 47M 31S
0          192.168.0.3      1.0.0.3                     Spoke  No   0H 28M 25S
0          192.168.0.4      1.0.0.4                     Spoke  No   0H 19M 15S
```

The output shows that Hub 1, Hub 2, Spoke 1, and Spoke 2 have all registered their address mapping information with the VAM servers.

# Display IPv4 ADVPN tunnel information on Hubs. This example uses Hub 1.

```
[Hub1] display advpn session
Interface         : Tunnel1
Number of sessions: 3
Private address  Public address              Port  Type  State      Holding time
192.168.0.2      1.0.0.2                     --    H-H   Success    0H 46M  8S
192.168.0.3      1.0.0.3                     --    H-S   Success    0H 27M 27S
192.168.0.4      1.0.0.4                     --    H-S   Success    0H 18M 18S
```

The output shows that Hub 1 has established a permanent tunnel to Hub 2, Spoke 1, and Spoke 2.

# Display IPv4 ADVPN tunnel information on Spokes. This example uses Spoke 1.

```
[Spoke1] display advpn session
Interface         : Tunnel1
```

```
Number of sessions: 2
Private address  Public address              Port  Type  State      Holding time
192.168.0.1      1.0.0.1                     --    S-H   Success    0H 46M  8S
192.168.0.2      1.0.0.2                     --    S-H   Success    0H 46M  8S
```

The output shows that Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.

# Verify that Spoke 1 can ping the private address 192.168.0.4 of Spoke 2.

```
[Spoke1] ping 192.168.0.4
Ping 192.168.0.4 (192.168.0.4): 56 data bytes, press CTRL_C to break
56 bytes from 192.168.0.4: icmp_seq=0 ttl=255 time=4.000 ms
56 bytes from 192.168.0.4: icmp_seq=1 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 192.168.0.4: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 192.168.0.4 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.000/4.000/1.549 ms
```

Example: Configuring IPv6 hub-spoke ADVPN

Network configuration

As shown in Figure 10, the primary and secondary VAM servers manage and maintain VAM client information for all hubs and spokes. The AAA server performs authentication and accounting for VAM clients. The two hubs back up each other, and perform data forwarding and route exchange.

Establish a permanent ADVPN tunnel between each spoke and each hub.

Figure 10 Network diagram

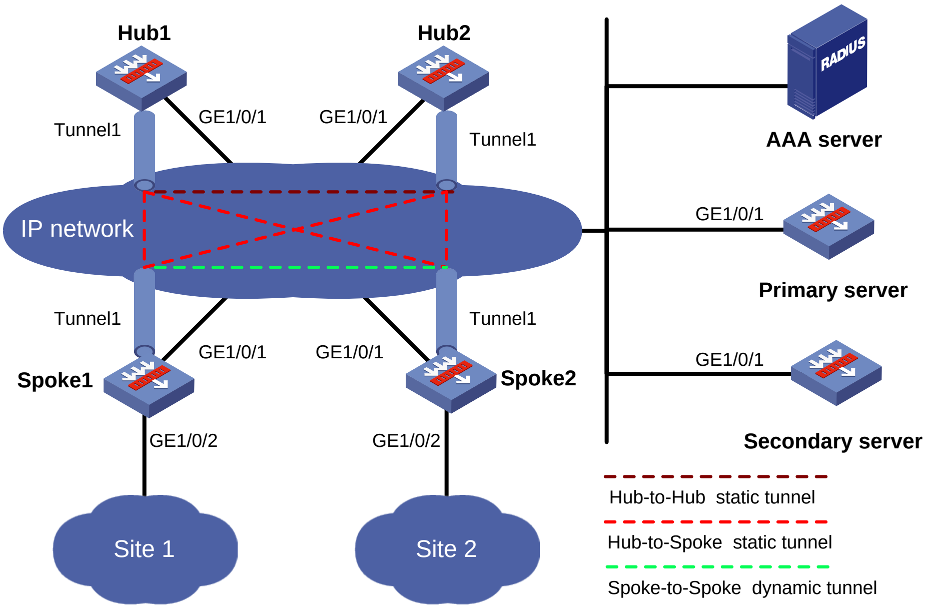


Table 4 Interface and IP address assignment

Device	Interfac e	IP address	Device	Interface	IP address
Hub 1	GE1/0/1	1::1/64	Spoke 1	GE1/0/1	1::3/64
	Tunnel1	192:168::1/64		GE1/0/2	192:168:1::1/64
Hub 2	GE1/0/1	1::2/64		Tunnel1	192:168::3/64
	Tunnel1	192:168::2/64	Spoke 2	GE1/0/1	1::4/64
AAA server		1::10/64		GE1/0/2	192:168:2::1/64
Primary server	GE1/0/1	1::11/64		Tunnel1	192:168::4/64
Secondary server	GE1/0/1	1::12/64			

Configuring the primary VAM server

1. Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

```
<PrimaryServer> system-view
[PrimaryServer] interface gigabitethernet 1/0/1
[PrimaryServer-GigabitEthernet1/0/1] ipv6 address 1::11/64
[PrimaryServer-GigabitEthernet1/0/1] quit
```

2. Add the interface to a security zone.

```
[PrimaryServer] security-zone name untrust
[PrimaryServer-security-zone-Untrust] import interface gigabitethernet 1/0/1
[PrimaryServer-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow the VAM server to send VAM protocol packets to the VAM clients.

```
[PrimaryServer] security-policy ipv6
[PrimaryServer-security-policy-ipv6] rule name advpnlocalout
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] source-zone local
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::11
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::1
1::12
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] action pass
[PrimaryServer-security-policy-ipv6-1-advpnlocalout] quit
```

# Configure a rule named **advpnlocalin** to allow the VAM server to receive VAM protocol packets from the VAM clients.

```
[PrimaryServer-security-policy-ipv6] rule name advpnlocalin
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] destination-zone local
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::1 1::12
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::11
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] action pass
[PrimaryServer-security-policy-ipv6-2-advpnlocalin] quit
[PrimaryServer-security-policy-ipv6] quit
```

4. Configure AAA:

# Configure RADIUS scheme **abc**.

```
<PrimaryServer> system-view
```

```
[PrimaryServer] radius scheme abc
[PrimaryServer-radius-abc] primary authentication ipv6 1::10 1812
[PrimaryServer-radius-abc] primary accounting ipv6 1::10 1813
[PrimaryServer-radius-abc] key authentication simple 123
[PrimaryServer-radius-abc] key accounting simple 123
[PrimaryServer-radius-abc] user-name-format without-domain
[PrimaryServer-radius-abc] quit
[PrimaryServer] radius session-control enable
```

# Configure AAA methods for ISP domain **abc**.

```
[PrimaryServer] domain abc
[PrimaryServer-isp-abc] authentication advpn radius-scheme abc
[PrimaryServer-isp-abc] accounting advpn radius-scheme abc
[PrimaryServer-isp-abc] quit
[PrimaryServer] domain default enable abc
```

5. Configure the VAM server:

# Create ADVPN domain **abc**.

```
[PrimaryServer] vam server advpn-domain abc id 1
```

# Create hub group 0.

```
[PrimaryServer-vam-server-domain-abc] hub-group 0
```

# Specify hub private IPv6 addresses.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub ipv6 private-address
192:168::1
[PrimaryServer-vam-server-domain-abc-hub-group-0] hub ipv6 private-address
192:168::2
```

# Specify a spoke private IPv6 network.

```
[PrimaryServer-vam-server-domain-abc-hub-group-0] spoke ipv6 private-address
network 192:168::0 64
[PrimaryServer-vam-server-domain-abc-hub-group-0] quit
```

# Set the preshared key to **123456**.

```
[PrimaryServer-vam-server-domain-abc] pre-shared-key simple 123456
```

# Set the authentication mode to CHAP.

```
[PrimaryServer-vam-server-domain-abc] authentication-method chap
```

# Enable the VAM server for the ADVPN domain.

```
[PrimaryServer-vam-server-domain-abc] server enable
[PrimaryServer-vam-server-domain-abc] quit
```

Configuring the secondary VAM server

# Configure the secondary VAM server in the same way that the primary server is configured. (Details not shown.)

Configuring Hub 1

1. Assign IP addresses to interfaces:

# Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

```
<Hub1> system-view
[Hub1] interface gigabitethernet 1/0/1
[Hub1-GigabitEthernet1/0/1] ipv6 address 1::1 64
[Hub1-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Hub1] interface tunnel 1 mode advpn gre ipv6
[Hub1-Tunnel1] ipv6 address 192:168::1 64
[Hub1-Tunnel1] ipv6 address fe80::1 link-local
```

2. Add the interfaces to a security zone.

```
[Hub1] security-zone name untrust
[Hub1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Hub1-security-zone-Untrust] import interface tunnel1
[Hub1-security-zone-Untrust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Hub 1 to send VAM protocol packets to the VAM server.

```
[Hub1] security-policy ipv6
[Hub1-security-policy-ipv6] rule name advpnlocalout
[Hub1-security-policy-ipv6-1-advpnlocalout] source-zone local
[Hub1-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Hub1-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::1
[Hub1-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Hub1-security-policy-ipv6-1-ipseclocalout] action pass
[Hub1-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Hub 1 to receive VAM protocol packets from the VAM server.

```
[Hub1-security-policy-ipv6] rule name advpnlocalin
[Hub1-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Hub1-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Hub1-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Hub1-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::1
[Hub1-security-policy-ipv6-2-advpnlocalin] action pass
[Hub1-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Hub 1 to send IPsec negotiation packets to other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ipseclocalout
[Hub1-security-policy-ipv6-3-ipseclocalout] source-zone local
[Hub1-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Hub1-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::1
[Hub1-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::2 1::4
[Hub1-security-policy-ipv6-3-ipseclocalout] action pass
[Hub1-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 1 to receive IPsec negotiation packets from other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ipseclocalin
[Hub1-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Hub1-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Hub1-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::2 1::4
[Hub1-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::1
[Hub1-security-policy-ipv6-4-ipseclocalin] action pass
[Hub1-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 1 to send OSPF packets to other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ospflocalout
[Hub1-security-policy-ipv6-5-ospflocalout] source-zone local
[Hub1-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Hub1-security-policy-ipv6-5-ospflocalout] service ospf
[Hub1-security-policy-ipv6-5-ospflocalout] action pass
[Hub1-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 1 to receive OSPF packets from other VAM clients.

```
[Hub1-security-policy-ipv6] rule name ospflocalin
[Hub1-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Hub1-security-policy-ipv6-6-ospflocalin] destination-zone local
[Hub1-security-policy-ipv6-6-ospflocalin] service ospf
[Hub1-security-policy-ipv6-6-ospflocalin] action pass
[Hub1-security-policy-ipv6-6-ospflocalin] quit
```

4. Configure the VAM client:

# Create VAM client **Hub1**.

```
<Hub1> system-view
[Hub1] vam client name Hub1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub1-vam-client-Hub1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub1-vam-client-Hub1] pre-shared-key simple 123456
```

# Set the username and password to **hub1**.

```
[Hub1-vam-client-Hub1] user hub1 password simple hub1
```

# Specify the primary and secondary VAM servers.

```
[Hub1-vam-client-Hub1] server primary ipv6-address 1::11
[Hub1-vam-client-Hub1] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Hub1-vam-client-Hub1] client enable
[Hub1-vam-client-Hub1] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Hub1] ike keychain abc
[Hub1-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Hub1-ike-keychain-abc] quit
[Hub1] ike profile abc
[Hub1-ike-profile-abc] keychain abc
[Hub1-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Hub1] ipsec transform-set abc
[Hub1-ipsec-transform-set-abc] encapsulation-mode transport
[Hub1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Hub1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Hub1-ipsec-transform-set-abc] quit
[Hub1] ipsec profile abc isakmp
[Hub1-ipsec-profile-isakmp-abc] transform-set abc
```

```
[Hub1-ipsec-profile-isakmp-abc] ike-profile abc
[Hub1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Hub1] ospfv3 1
[Hub1-ospfv3-1] router-id 0.0.0.1
[Hub1-ospfv3-1] area 0
[Hub1-ospfv3-1-area-0.0.0.0] quit
[Hub1-ospfv3-1] quit
```

7. Configure interface Tunnel 1.

```
[Hub1] interface tunnel1
[Hub1-Tunnel1] vam ipv6 client Hub1
[Hub1-Tunnel1] ospfv3 1 area 0
[Hub1-Tunnel1] ospfv3 network-type p2mp
[Hub1-Tunnel1] source gigabitethernet 1/0/1
[Hub1-Tunnel1] tunnel protection ipsec profile abc
[Hub1-Tunnel1] quit
```

Configuring Hub 2

1. Assign IP addresses to interfaces:

   # Assign an IPv6 global unicast address to interface GigabitEthernet 1/0/1.

```
<Hub2> system-view
[Hub2] interface gigabitethernet 1/0/1
[Hub2-GigabitEthernet1/0/1] ipv6 address 1::2 64
[Hub2-GigabitEthernet1/0/1] quit
```

   # Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Hub2] interface tunnel 1 mode advpn gre ipv6
[Hub2-Tunnel1] ipv6 address 192:168::2 64
[Hub2-Tunnel1] ipv6 address fe80::2 link-local
```

2. Add the interfaces to a security zone.

```
[Hub2] security-zone name untrust
[Hub2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Hub2-security-zone-Untrust] import interface tunnel1
[Hub2-security-zone-Untrust] quit
```

3. Configure a security policy:

   # Configure a rule named **advpnlocalout** to allow Hub 2 to send VAM protocol packets to the VAM server.

```
[Hub2] security-policy ipv6
[Hub2-security-policy-ipv6] rule name advpnlocalout
[Hub2-security-policy-ipv6-1-advpnlocalout] source-zone local
[Hub2-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Hub2-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::2
[Hub2-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Hub2-security-policy-ipv6-1-ipseclocalout] action pass
[Hub2-security-policy-ipv6-1-ipseclocalout] quit
```

   # Configure a rule named **advpnlocalin** to allow Hub 2 to receive VAM protocol packets from the VAM server.

```
[Hub2-security-policy-ipv6] rule name advpnlocalin
```

```
[Hub2-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Hub2-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Hub2-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Hub2-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::2
[Hub2-security-policy-ipv6-2-advpnlocalin] action pass
[Hub2-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Hub 2 to send IPsec negotiation packets to other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ipseclocalout
[Hub2-security-policy-ipv6-3-ipseclocalout] source-zone local
[Hub2-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Hub2-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::2
[Hub2-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::4
[Hub2-security-policy-ipv6-3-ipseclocalout] action pass
[Hub2-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Hub 2 to receive IPsec negotiation packets from other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ipseclocalin
[Hub2-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Hub2-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Hub2-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::4
[Hub2-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::2
[Hub2-security-policy-ipv6-4-ipseclocalin] action pass
[Hub2-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Hub 2 to send OSPF packets to other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ospflocalout
[Hub2-security-policy-ipv6-5-ospflocalout] source-zone local
[Hub2-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Hub2-security-policy-ipv6-5-ospflocalout] service ospf
[Hub2-security-policy-ipv6-5-ospflocalout] action pass
[Hub2-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Hub 2 to receive OSPF packets from other VAM clients.

```
[Hub2-security-policy-ipv6] rule name ospflocalin
[Hub2-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Hub2-security-policy-ipv6-6-ospflocalin] destination-zone local
[Hub2-security-policy-ipv6-6-ospflocalin] service ospf
[Hub2-security-policy-ipv6-6-ospflocalin] action pass
[Hub2-security-policy-ipv6-6-ospflocalin] quit
[Hub2-security-policy-ipv6] quit
```

4. Configure the VAM client:

# Create VAM client **Hub2**.

```
<Hub2> system-view
[Hub2] vam client name Hub2
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Hub2-vam-client-Hub2] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Hub2-vam-client-Hub2] pre-shared-key simple 123456
```

# Set both the username and password to **hub2**.

```
[Hub2-vam-client-Hub2] user hub2 password simple hub2
```

# Specify the primary and secondary VAM servers.

```
[Hub2-vam-client-Hub2] server primary ipv6-address 1::11
[Hub2-vam-client-Hub2] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Hub2-vam-client-Hub2] client enable
[Hub2-vam-client-Hub2] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Hub2] ike keychain abc
[Hub2-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Hub2-ike-keychain-abc] quit
[Hub2] ike profile abc
[Hub2-ike-profile-abc] keychain abc
[Hub2-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Hub2] ipsec transform-set abc
[Hub2-ipsec-transform-set-abc] encapsulation-mode transport
[Hub2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Hub2-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Hub2-ipsec-transform-set-abc] quit
[Hub2] ipsec profile abc isakmp
[Hub2-ipsec-profile-isakmp-abc] transform-set abc
[Hub2-ipsec-profile-isakmp-abc] ike-profile abc
[Hub2-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Hub2] ospfv3 1
[Hub2-ospfv3-1] router-id 0.0.0.2
[Hub2-ospfv3-1] area 0
[Hub2-ospfv3-1-area-0.0.0.0] quit
[Hub2-ospfv3-1] quit
```

7. Configure interface Tunnel 1.

```
[Hub2] interface tunnel1
[Hub2-Tunnel1] vam ipv6 client Hub2
[Hub2-Tunnel1] ospfv3 1 area 0
[Hub2-Tunnel1] ospfv3 network-type p2mp
[Hub2-Tunnel1] source gigabitethernet 1/0/1
[Hub2-Tunnel1] tunnel protection ipsec profile abc
[Hub2-Tunnel1] quit
```

Configuring Spoke 1

1. Assign IP addresses to interfaces:

# Assign a global unicast address to interface GigabitEthernet 1/0/1.

```
<Spoke1> system-view
[Spoke1] interface gigabitethernet 1/0/1
[Spoke1-GigabitEthernet1/0/1] ipv6 address 1::3 64
```

```
[Spoke1-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Spoke1] interface tunnel 1 mode advpn gre ipv6
[Spoke1-Tunnel1] ipv6 address 192:168::3 64
[Spoke1-Tunnel1] ipv6 address fe80::3 link-local
```

# Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke1] security-zone name untrust
[Spoke1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Spoke1-security-zone-Untrust] import interface tunnel1
[Spoke1-security-zone-Untrust] quit
[Spoke1] security-zone name trust
[Spoke1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke1-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Spoke 1 to send VAM protocol packets to the VAM server.

```
[Spoke1] security-policy ipv6
[Spoke1-security-policy-ipv6] rule name advpnlocalout
[Spoke1-security-policy-ipv6-1-advpnlocalout] source-zone local
[Spoke1-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::3
[Spoke1-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Spoke1-security-policy-ipv6-1-ipseclocalout] action pass
[Spoke1-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Spoke 1 to receive VAM protocol packets from the VAM server.

```
[Spoke1-security-policy-ipv6] rule name advpnlocalin
[Spoke1-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Spoke1-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Spoke1-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Spoke1-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::3
[Spoke1-security-policy-ipv6-2-advpnlocalin] action pass
[Spoke1-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 1 to send IPsec negotiation packets to other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ipseclocalout
[Spoke1-security-policy-ipv6-3-ipseclocalout] source-zone local
[Spoke1-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::3
[Spoke1-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::4
[Spoke1-security-policy-ipv6-3-ipseclocalout] action pass
[Spoke1-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 1 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ipseclocalin
[Spoke1-security-policy-ipv6-4-ipseclocalin] source-zone untrust
```

```
[Spoke1-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Spoke1-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::4
[Spoke1-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::3
[Spoke1-security-policy-ipv6-4-ipseclocalin] action pass
[Spoke1-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 1 to send OSPF packets to other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ospflocalout
[Spoke1-security-policy-ipv6-5-ospflocalout] source-zone local
[Spoke1-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Spoke1-security-policy-ipv6-5-ospflocalout] service ospf
[Spoke1-security-policy-ipv6-5-ospflocalout] action pass
[Spoke1-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 1 to receive OSPF packets from other VAM clients.

```
[Spoke1-security-policy-ipv6] rule name ospflocalin
[Spoke1-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Spoke1-security-policy-ipv6-6-ospflocalin] destination-zone local
[Spoke1-security-policy-ipv6-6-ospflocalin] service ospf
[Spoke1-security-policy-ipv6-6-ospflocalin] action pass
[Spoke1-security-policy-ipv6-6-ospflocalin] quit
[Spoke1-security-policy-ipv6] quit
```

4. Configure the VAM client:

# Create VAM client **Spoke1**.

```
<Spoke1> system-view
[Spoke1] vam client name Spoke1
```

# Specify ADVPN domain **abc** for the VAM client.

```
[Spoke1-vam-client-Spoke1] advpn-domain abc
```

# Set the preshared key to **123456**.

```
[Spoke1-vam-client-Spoke1] pre-shared-key simple 123456
```

# Set both the username and password to **spoke1**.

```
[Spoke1-vam-client-Spoke1] user spoke1 password simple spoke1
```

# Specify the primary and secondary VAM servers.

```
[Spoke1-vam-client-Spoke1] server primary ipv6-address 1::11
[Spoke1-vam-client-Spoke1] server secondary ipv6-address 1::12
```

# Enable the VAM client.

```
[Spoke1-vam-client-Spoke1] client enable
[Spoke1-vam-client-Spoke1] quit
```

5. Configure an IPsec profile:

# Configure IKE.

```
[Spoke1] ike keychain abc
[Spoke1-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
[Spoke1-ike-keychain-abc] quit
[Spoke1] ike profile abc
[Spoke1-ike-profile-abc] keychain abc
[Spoke1-ike-profile-abc] quit
```

# Configure the IPsec profile.

```
[Spoke1] ipsec transform-set abc
```

```
[Spoke1-ipsec-transform-set-abc] encapsulation-mode transport
[Spoke1-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
[Spoke1-ipsec-transform-set-abc] esp authentication-algorithm sha1
[Spoke1-ipsec-transform-set-abc] quit
[Spoke1] ipsec profile abc isakmp
[Spoke1-ipsec-profile-isakmp-abc] transform-set abc
[Spoke1-ipsec-profile-isakmp-abc] ike-profile abc
[Spoke1-ipsec-profile-isakmp-abc] quit
```

6. Configure OSPFv3.

```
[Spoke1] ospfv3 1
[Spoke1-ospfv3-1] router-id 0.0.0.3
[Spoke1-ospfv3-1] area 0
[Spoke1-ospfv3-1-area-0.0.0.0] quit
[Spoke1-ospfv3-1] quit
```

7. Configure interface Tunnel 1.

```
[Spoke1] interface tunnel1
[Spoke1-Tunnel1] vam ipv6 client Spoke1
[Spoke1-Tunnel1] ospfv3 1 area 0
[Spoke1-Tunnel1] ospfv3 network-type p2mp
[Spoke1-Tunnel1] source gigabitethernet 1/0/1
[Spoke1-Tunnel1] tunnel protection ipsec profile abc
[Spoke1-Tunnel1] quit
```

Configuring Spoke 2

1. Assign IP addresses to interfaces:

# Assign a global unicast address to interface GigabitEthernet 1/0/1.

```
<Spoke2> system-view
[Spoke2] interface gigabitethernet 1/0/1
[Spoke2-GigabitEthernet1/0/1] ipv6 address 1::4 64
[Spoke2-GigabitEthernet1/0/1] quit
```

# Create interface Tunnel 1 and set its tunnel mode to the GRE-encapsulated IPv6 ADVPN tunnel mode. Then, assign an IPv6 global unicast addresses and an IPv6 link-local address to the interface.

```
[Spoke2] interface tunnel 1 mode advpn gre ipv6
[Spoke2-Tunnel1] ipv6 address 192:168::4 64
[Spoke2-Tunnel1] ipv6 address fe80::4 link-local
```

# Assign IP addresses to other interfaces. (Details not shown.)

2. Add the interfaces to security zones.

```
[Spoke2] security-zone name untrust
[Spoke2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Spoke2-security-zone-Untrust] import interface tunnel1
[Spoke2-security-zone-Untrust] quit
[Spoke2] security-zone name trust
[Spoke2-security-zone-Trust] import interface gigabitethernet 1/0/2
[Spoke2-security-zone-Trust] quit
```

3. Configure a security policy:

# Configure a rule named **advpnlocalout** to allow Spoke 2 to send VAM protocol packets to the VAM server.

```
[Spoke2] security-policy ipv6
```

```
[Spoke2-security-policy-ipv6] rule name advpnlocalout
[Spoke2-security-policy-ipv6-1-advpnlocalout] source-zone local
[Spoke2-security-policy-ipv6-1-advpnlocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-1-advpnlocalout] source-ip-host 1::4
[Spoke2-security-policy-ipv6-1-advpnlocalout] destination-ip-range 1::10 1::12
[Spoke2-security-policy-ipv6-1-ipseclocalout] action pass
[Spoke2-security-policy-ipv6-1-ipseclocalout] quit
```

# Configure a rule named **advpnlocalin** to allow Spoke 2 to receive VAM protocol packets from the VAM server.

```
[Spoke2-security-policy-ipv6] rule name advpnlocalin
[Spoke2-security-policy-ipv6-2-advpnlocalin] source-zone untrust
[Spoke2-security-policy-ipv6-2-advpnlocalin] destination-zone local
[Spoke2-security-policy-ipv6-2-advpnlocalin] source-ip-range 1::10 1::12
[Spoke2-security-policy-ipv6-2-advpnlocalin] destination-ip-host 1::4
[Spoke2-security-policy-ipv6-2-advpnlocalin] action pass
[Spoke2-security-policy-ipv6-2-advpnlocalin] quit
```

# Configure a rule named **ipseclocalout** to allow Spoke 2 to send IPsec negotiation packets to other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ipseclocalout
[Spoke2-security-policy-ipv6-3-ipseclocalout] source-zone local
[Spoke2-security-policy-ipv6-3-ipseclocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-3-ipseclocalout] source-ip-host 1::4
[Spoke2-security-policy-ipv6-3-ipseclocalout] destination-ip-range 1::1 1::3
[Spoke2-security-policy-ipv6-3-ipseclocalout] action pass
[Spoke2-security-policy-ipv6-3-ipseclocalout] quit
```

# Configure a rule named **ipseclocalin** to allow Spoke 2 to receive IPsec negotiation packets from other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ipseclocalin
[Spoke2-security-policy-ipv6-4-ipseclocalin] source-zone untrust
[Spoke2-security-policy-ipv6-4-ipseclocalin] destination-zone local
[Spoke2-security-policy-ipv6-4-ipseclocalin] source-ip-range 1::1 1::3
[Spoke2-security-policy-ipv6-4-ipseclocalin] destination-ip-host 1::4
[Spoke2-security-policy-ipv6-4-ipseclocalin] action pass
[Spoke2-security-policy-ipv6-4-ipseclocalin] quit
```

# Configure a rule named **ospflocalout** to allow Spoke 2 to send OSPF packets to other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ospflocalout
[Spoke2-security-policy-ipv6-5-ospflocalout] source-zone local
[Spoke2-security-policy-ipv6-5-ospflocalout] destination-zone untrust
[Spoke2-security-policy-ipv6-5-ospflocalout] service ospf
[Spoke2-security-policy-ipv6-5-ospflocalout] action pass
[Spoke2-security-policy-ipv6-5-ospflocalout] quit
```

# Configure a rule named **ospflocalin** to allow Spoke 2 to receive OSPF packets from other VAM clients.

```
[Spoke2-security-policy-ipv6] rule name ospflocalin
[Spoke2-security-policy-ipv6-6-ospflocalin] source-zone untrust
[Spoke2-security-policy-ipv6-6-ospflocalin] destination-zone local
[Spoke2-security-policy-ipv6-6-ospflocalin] service ospf
[Spoke2-security-policy-ipv6-6-ospflocalin] action pass
```

```
[Spoke2-security-policy-ipv6-6-ospflocalin] quit
[Spoke2-security-policy-ipv6] quit
```

4. Configure the VAM client:

   # Create VAM client **Spoke2**.

   ```
   <Spoke2> system-view
   [Spoke2] vam client name Spoke2
   ```

   # Specify ADVPN domain **abc** for the VAM client.

   ```
   [Spoke2-vam-client-Spoke2] advpn-domain abc
   ```

   # Set the preshared key to **123456**.

   ```
   [Spoke2-vam-client-Spoke2] pre-shared-key simple 123456
   ```

   # Set both the username and password to **spoke2**.

   ```
   [Spoke2-vam-client-Spoke2] user spoke2 password simple spoke2
   ```

   # Specify the primary and secondary VAM servers.

   ```
   [Spoke2-vam-client-Spoke2] server primary ipv6-address 1::11
   [Spoke2-vam-client-Spoke2] server secondary ipv6-address 1::12
   ```

   # Enable the VAM client.

   ```
   [Spoke2-vam-client-Spoke2] client enable
   [Spoke2-vam-client-Spoke2] quit
   ```

5. Configure an IPsec profile:

   # Configure IKE.

   ```
   [Spoke2] ike keychain abc
   [Spoke2-ike-keychain-abc] pre-shared-key address ipv6 :: 0 key simple 123456
   [Spoke2-ike-keychain-abc] quit
   [Spoke2] ike profile abc
   [Spoke2-ike-profile-abc] keychain abc
   [Spoke2-ike-profile-abc] quit
   ```

   # Configure the IPsec profile.

   ```
   [Spoke2] ipsec transform-set abc
   [Spoke2-ipsec-transform-set-abc] encapsulation-mode transport
   [Spoke2-ipsec-transform-set-abc] esp encryption-algorithm des-cbc
   [Spoke2-ipsec-transform-set-abc] esp authentication-algorithm sha1
   [Spoke2-ipsec-transform-set-abc] quit
   [Spoke2] ipsec profile abc isakmp
   [Spoke2-ipsec-profile-isakmp-abc] transform-set abc
   [Spoke2-ipsec-profile-isakmp-abc] ike-profile abc
   [Spoke2-ipsec-profile-isakmp-abc] quit
   ```

6. Configure OSPFv3.

   ```
   [Spoke2] ospfv3 1
   [Spoke2-ospfv3-1] router-id 0.0.0.4
   [Spoke2-ospfv3-1] area 0
   [Spoke2-ospfv3-1-area-0.0.0.0] quit
   [Spoke2-ospfv3-1] quit
   ```

7. Configure interface Tunnel 1.

   ```
   [Spoke2] interface tunnel1
   [Spoke2-Tunnel1] vam ipv6 client Spoke2
   [Spoke2-Tunnel1] ospfv3 1 area 0
   [Spoke2-Tunnel1] ospfv3 network-type p2mp
   ```

```
    [Spoke2-Tunnel1] source gigabitethernet 1/0/1
    [Spoke2-Tunnel1] tunnel protection ipsec profile abc
    [Spoke2-Tunnel1] quit
```

Verifying the configuration

# Display IPv6 address mapping information for all VAM clients registered with the primary VAM server.

```
[PrimaryServer] display vam server ipv6 address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address       Public address         Type   NAT  Holding time
0          192:168::1            1::1                   Hub    No   0H 52M  7S
0          192:168::2            1::2                   Hub    No   0H 47M 31S
0          192:168::3            1::3                   Spoke  No   0H 28M 25S
0          192:168::4            1::4                   Spoke  No   0H 19M 15S
```

# Display IPv6 address mapping information for all VAM clients registered with the secondary VAM server.

```
[SecondaryServer] display vam server ipv6 address-map
ADVPN domain name: abc
Total private address mappings: 4
Group      Private address       Public address         Type   NAT  Holding time
0          192:168::1            1::1                   Hub    No   0H 52M  7S
0          192:168::2            1::2                   Hub    No   0H 47M 31S
0          192:168::3            1::3                   Spoke  No   0H 28M 25S
0          192:168::4            1::4                   Spoke  No   0H 19M 15S
```

The output shows that Hub 1, Hub 2, Spoke 1, and Spoke 2 have all registered their address mapping information with the VAM servers.

# Display IPv6 ADVPN tunnel information on Hubs. This example uses Hub 1.

```
[Hub1] display advpn ipv6 session
Interface         : Tunnel1
Number of sessions: 3
Private address       Public address        Port  Type  State      Holding time
192:168::2            1::2                  --    H-H   Success    0H 46M  8S
192:168::3            1::3                  --    H-S   Success    0H 27M 27S
192:168::4            1::4                  --    H-S   Success    0H 18M 18S
```

The output shows that Hub 1 has established a permanent tunnel to Hub 2, Spoke 1, and Spoke 2.

# Display IPv6 ADVPN tunnel information on Spokes. This example uses Spoke 1.

```
[Spoke1] display advpn ipv6 session
Interface         : Tunnel1
Number of sessions: 2
Private address       Public address        Port  Type  State      Holding time
192:168::1            1::1                  --    S-H   Success    0H 46M  8S
192:168::2            1::2                  --    S-H   Success    0H 46M  8S
```

The output shows that Spoke 1 has established a permanent hub-spoke tunnel to Hub 1 and Hub 2.

# Verify that Spoke 1 can ping the private address 192:168::4 of Spoke 2.

```
[Spoke1] ping ipv6 192:168::4
Ping6(56 data bytes) 192:168::4 --> 192:168::4, press CTRL_C to break
56 bytes from 192:168::4, icmp_seq=0 hlim=64 time=3.000 ms
```

```
56 bytes from 192:168::4, icmp_seq=1 hlim=64 time=0.000 ms
56 bytes from 192:168::4, icmp_seq=2 hlim=64 time=1.000 ms
56 bytes from 192:168::4, icmp_seq=3 hlim=64 time=1.000 ms
56 bytes from 192:168::4, icmp_seq=4 hlim=64 time=1.000 ms

--- Ping6 statistics for 192:168::4 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/1.200/3.000/0.980 ms
```

NSFOCUS Firewall Series

NF Internet Access Behavior Management Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for Internet access behavior management features, including:bandwidth management, application audit and management and NetShare control.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

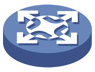
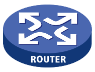
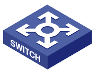
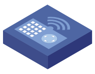
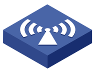
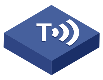
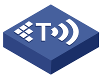
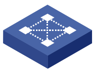
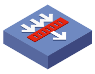
GUI conventions

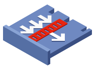
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring bandwidth management

About bandwidth management

Bandwidth management provides fine-grained control over traffic that flows through the device by using the following information:

- Source and destination security zones.
- Source and destination IP addresses.
- Services.
- Users/user groups.
- Applications.
- DSCP priorities.

Application scenario

Bandwidth management is used in the following scenarios:

- Enterprise intranet users need far more bandwidth than the amount of bandwidth leased from an ISP. This creates a bandwidth bottleneck at the intranet egress.
- The P2P traffic on the intranet egress consumes a majority of the bandwidth resources. As a result, bandwidth cannot be guaranteed for key services.

Bandwidth management allows you to deploy traffic rules on the network egress for different traffic types. Bandwidth management improves bandwidth efficiency and guarantees bandwidth for key services when congestion occurs.

Bandwidth management process

Bandwidth management is implemented through the traffic policy. You can configure traffic profiles and traffic rules in traffic policy view. A traffic profile specifies the guaranteed bandwidth and maximum bandwidth. A traffic rule specifies match criteria to match packets and the traffic profile to apply to matching packets.

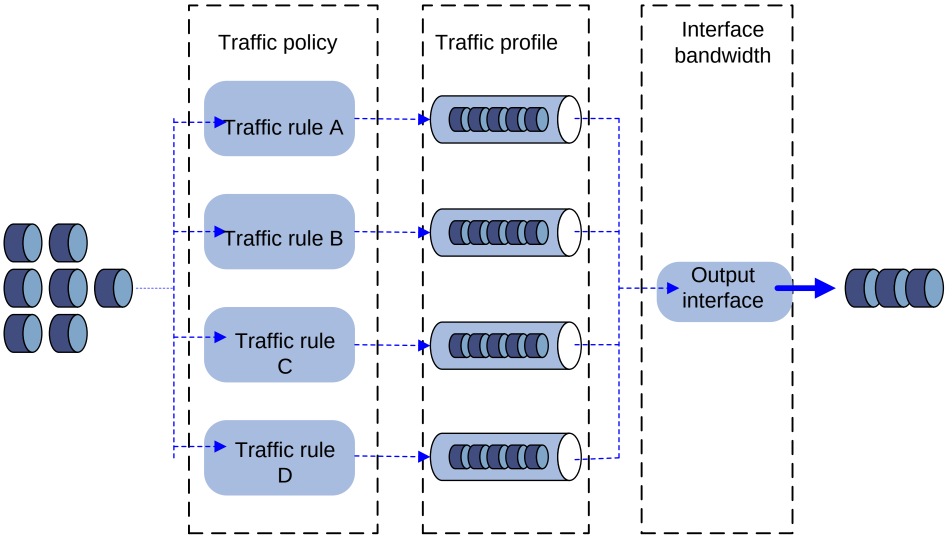
As shown in Figure 1, the bandwidth management process is as follows:

1. The device matches the packet against the match criteria in a traffic rule.

   The packet meets a match criterion if it matches any of its match values. A packet does not match a match criterion if it matches none of its match values.
2. If the packet meets all match criteria in the traffic rule (for the user and user group criteria or application and application group criteria, only one criterion needs to be matched), the packet matches the traffic rule. Otherwise, the packet does not match the traffic rule and continues to be matched by the next traffic rule. If the packet does not match any traffic rule, the packet is forwarded without bandwidth management.
3. After the packet matches a traffic rule, the interface processes the packet according to the traffic profile (if any) specified for the traffic rule.

   If no traffic profile is specified for the traffic rule, the packet is forwarded without bandwidth management.
4. The traffic profile processes the packet according to its settings.
5. If the interface is configured with a QoS feature in the outbound direction, the interface performs bandwidth management before performing QoS.
6. The packet is controlled by the interface bandwidth of the output interface.

Figure 1 Bandwidth management process



Traffic rule

Multiple traffic rules can be configured in the traffic policy. For a traffic rule, you can define the match criteria to match packets and specify the traffic profile to apply to matching packets.

Traffic rules support rule nesting, which allows a traffic rule to have a parent traffic rule. A maximum of four nesting levels are supported.

Match criteria in a traffic rule

A traffic rule can have multiple match criteria. You can configure the following match criteria in a traffic rule:

- Source and destination security zones.
- Source and destination IP addresses.
- Services.
- Users/user groups.
- Applications.
- DSCP priorities.

One match criterion can contain multiple match values. For example, you can configure multiple applications for an application match criterion.

Action in a traffic rule

You can use a traffic profile for an action in a traffic rule. The device limits the matching traffic according to the settings in the traffic profile.

Match order for parent and child traffic rules

The following rules apply when the device matches a traffic rule with a parent traffic rule:

- The parent traffic rule is first matched. After the parent traffic rule is matched, the child traffic rule is matched. If the parent traffic rule is not matched, the child traffic rule is ignored and the matching process fails.
- If both parent and child traffic rules are matched, the traffic profile for the child traffic rule is executed before the traffic profile for the parent traffic rule is executed. If both parent and child traffic rules are about the same parameter, the smaller value for an upper-limit parameter or the

larger value for a lower-limit parameter is applied. If only the parent traffic rule is matched, the traffic profile for the parent traffic rule is applied.

Traffic profile

A traffic profile defines bandwidth resources that can be used by a traffic type. The interface bandwidth can be allocated among multiple traffic profiles. You can configure the following bandwidth limit parameters and priority parameters in a traffic profile:

Rate limit mode for a traffic profile

You can limit the traffic rate in one of the following ways:

- Limit the upstream bandwidth and downstream bandwidth separately.
- Limit the upstream bandwidth and downstream bandwidth as a whole.

Total bandwidth limits

- **Total guaranteed bandwidth**—Guarantees the total minimum bandwidth for key services when congestion occurs.
- **Total maximum bandwidth**—Controls the total maximum bandwidth for non-key services to prevent them consuming a large amount of bandwidth.

Per-IP or per-user bandwidth limits

- **Per-IP or per-user guaranteed bandwidth**—Guarantees the minimum bandwidth per IP address or per user to provide for bandwidth management at finer granularity.
- **Per-IP or per-user maximum bandwidth**—Controls the maximum bandwidth allowed per IP address or per user to provide for bandwidth management at finer granularity.

Per-rule, per-IP, or per-user connection limits

- **Per-rule, per-IP, or per-user connection limits**—You can set the connection count limit and connection rate limit to prevent the following situations:
  - The system resources on the device are exhausted because internal users initiate a large number of connections to external networks in a short time period.
  - An internal server cannot process normal connection requests because it receives a large number of connection requests in a short time period.

Priority parameters

- **Traffic priority**—When an interface is congested with packets of multiple traffic profiles, packets with higher priority are sent first. Packets with the same priority have the same chance of being forwarded.
- **DSCP marking**—Modifies the DSCP value in packets. Network devices can classify traffic by using DSCP values and provide different treatment for packets according to the modified DSCP values.

Restrictions and guidelines: Bandwidth management configuration

When you configure bandwidth management, follow these restrictions and guidelines:

- As a best practice, observe the depth-first principle when creating policies. Always create a policy with a smaller management scope before a policy with a larger management scope.
- An interface with small default expected bandwidth might experience traffic loss if the following conditions exist:
  - There is a large amount of traffic on the interface.

  - The interface uses the default expected bandwidth.

  To avoid traffic loss, implicitly set the expected bandwidth to a large value for such an interface. For example, you can set the expected bandwidth of a tunnel interface to a value greater than 64 kbps (the default) if there is a large amount of traffic on the interface.

Prerequisites for bandwidth management

Before configuring bandwidth management, complete the following tasks:

- Configure time ranges (see time range configuration in *ACL and QoS Configuration Guide*).
- Configure IP address object groups and service object groups (see object group configuration in *Security Configuration Guide*).
- Configure applications (see APR configuration in *Security Configuration Guide*).
- Configure users and user groups (see user identification configuration in *Security Configuration Guide*).
- Configure security zones (see security zone configuration in *Security Configuration Guide*).

Bandwidth management tasks at a glance

To configure bandwidth management, perform the following tasks:

1. Configuring a traffic profile
   - Creating a traffic profile
   - Configuring bandwidth limits for the traffic profile
   - (Optional.) Configuring bandwidth detection for the traffic profile
   - Setting the reference mode for the traffic profile
   - (Optional.) Renaming the traffic profile
2. Configuring a traffic rule
   - Creating a traffic rule
   - Configuring match criteria for the traffic rule
   - Specifying an action for the traffic rule
   - (Optional.) Specifying a time range for the traffic rule
3. (Optional.) Managing and maintaining a traffic rule
   - Copying a traffic rule
   - Renaming a traffic rule
   - Moving a traffic rule
   - Disabling a traffic rule
4. (Optional.) Enabling bandwidth management statistics collection

Configuring a traffic profile

Creating a traffic profile

1. Enter system view.

   `system-view`
2. Enter traffic policy view.

   `traffic-policy`

3. Create a traffic profile and enter traffic profile view.

   profile name *profile-name*

Configuring bandwidth limits for the traffic profile

About this task

A traffic profile defines the bandwidth resources that can be used and takes effect after it is specified for a traffic rule.

Restrictions and guidelines

- Any two of the following settings are mutually exclusive:
  - Per-IP maximum bandwidth.
  - Per-user maximum bandwidth.
  - Dynamic and even allocation for maximum bandwidth.

  The most recent configuration takes effect.
- The per-IP guaranteed bandwidth setting and per-user guaranteed bandwidth setting are mutually exclusive.

Procedure

1. Enter system view.

   system-view
2. Enter traffic policy view.

   traffic-policy
3. Enter traffic profile view.

   profile name *profile-name*
4. Configure bandwidth settings.
   - Set the total guaranteed bandwidth or maximum bandwidth for the traffic profile.

     bandwidth { **downstream** | **total** | **upstream** } { **guaranteed** | **maximum** } *bandwidth-value*

     By default, the total guaranteed bandwidth and maximum bandwidth are not set.

     The maximum bandwidth must be greater than or equal to the guaranteed bandwidth.

     Before you can enable dynamic and even allocation for maximum bandwidth, you must set the total maximum bandwidth.
   - Set the per-IP or per-user guaranteed bandwidth or maximum bandwidth for the traffic profile.

     bandwidth { **downstream** | **total** | **upstream** } { **guaranteed** | **maximum** } { **per-ip** | **per-user** } *bandwidth-value*

     By default, the per-IP or per-user guaranteed bandwidth and maximum bandwidth are not set.
   - Set the TCP MSS for the traffic profile.

     tcp mss *mss-value*

     By default, the TCP MSS is not set.
5. Set the per-IP monthly traffic quota.

   bandwidth total traffic-quota per-ip monthly *quota-value*

   By default, the amount of traffic used by an IP address per month is not limited.
6. Enable dynamic and even allocation for maximum bandwidth.

   bandwidth average enable

By default, dynamic and even allocation for maximum bandwidth is disabled.

7. Configure connection limit settings.
   - Set the connection count limit for the traffic profile.

     connection-limit count { **per-rule** | **per-ip** | **per-user** } *connection-number*

     By default, the connection count limit is not set.
   - Set the connection rate limit for the traffic profile.

     connection-limit rate { **per-rule** | **per-ip** | **per-user** } *connection-rate*

     By default, the connection rate limit is not set.
8. Configure priority settings.
   - Set the traffic priority for packets of the traffic profile.

     traffic-priority *priority-value*

     By default, the traffic priority for packets of a traffic profile is 1.
   - Mark the DSCP value for packets of the traffic profile.

     remark dscp *dscp-value*

     By default, the DSCP value for packets of a traffic profile is not marked.

Configuring bandwidth detection for the traffic profile

About this task

This feature monitors the traffic rates based on source IP addresses in real time to identify the maximum rate and minimum rate of each IP address. If the traffic rate of an IP address exceeds or falls below a user-configured bandwidth threshold, the device sends logs to the log host by using the fast log output feature.

You can configure static bandwidth thresholds or configure the dynamic bandwidth threshold learning feature.

- **Static bandwidth threshold**—Allows you to configure a minimum threshold and a maximum threshold.
- **Dynamic threshold learning**—Allows the device to obtain minimum and maximum bandwidth thresholds by dynamically learning traffic rates. This feature is useful if you do not know the traffic patterns in a network and cannot determine appropriate bandwidth thresholds. With this feature enabled, the device measures the traffic rates over a user-configured duration and calculates an average rate. Then, the device obtains the minimum and maximum bandwidth thresholds by using the average rate multiplied by the minimum and maximum tolerance values.

If you configure both static bandwidth thresholds and the dynamic bandwidth threshold learning feature for the traffic profile, the following rules apply:

- Before the device learns the average traffic rate, it uses the static bandwidth thresholds.
- After the device learns the average traffic rate, it uses the dynamic bandwidth thresholds.

Procedure

1. Enter system view.

   system-view
2. Enter traffic policy view.

   traffic-policy
3. Enter traffic profile view.

   profile name *profile-name*

4. Enable per-IP bandwidth threshold detection.

   per-ip bandwidth-threshold-detect enable

   By default, per-IP threshold bandwidth detection is disabled.

5. Configure per-IP static bandwidth thresholds.
   - Set the maximum bandwidth threshold.

     per-ip bandwidth-threshold max-value *max-value*

     By default, the maximum bandwidth threshold is not set.
   - Set the minimum bandwidth threshold.

     per-ip bandwidth-threshold min-value *min-value*

     By default, the maximum bandwidth threshold is not set.

6. Configure per-IP dynamic bandwidth threshold learning.
   a. Enable per-IP dynamic bandwidth threshold learning.

      per-ip bandwidth-threshold-learn enable

      By default, per-IP dynamic bandwidth threshold learning is disabled.

   b. Set the duration for per-IP dynamic bandwidth threshold learning.

      per-ip bandwidth-threshold-learn duration *duration-value*

      By default, the duration for per-IP dynamic bandwidth threshold learning is 1440 minutes (24 hours).

      As a best practice, set the learning duration to be longer than 1440 minutes for the device to learn traffic for no less than a whole day.

   c. Set the maximum tolerance value.

      per-ip bandwidth-threshold-learn tolerance max-value *max-value*

      By default, the maximum tolerance value is not set.

   d. Set the minimum tolerance value.

      per-ip bandwidth-threshold-learn tolerance m min-value *min-value*

      By default, the minimum tolerance value is not set.

Setting the reference mode for the traffic profile

About this task

A traffic profile can be referenced by multiple traffic rules in one of the following ways:

- **per-rule**—Each rule that uses the profile can reach the bandwidth limits and connection limits specified in the profile.
- **rule-shared**—All rules that use the profile share the bandwidth limits and connection limits specified in the profile.

Procedure

1. Enter system view.

   system-view

2. Enter traffic policy view.

   traffic-policy

3. Enter traffic profile view.

   profile name *profile-name*

4. Set the reference mode for the traffic profile.

   profile reference-mode { **per-rule** | **rule-shared** }

The default setting is **per-rule**.

Renaming the traffic profile

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Rename a traffic profile.
   profile rename *old-name new-name*

Configuring a traffic rule

Creating a traffic rule

About this task

For a new traffic rule to inherit the match criteria of an existing traffic rule, specify the existing traffic rule as the parent of the new traffic rule. You can specify traffic profiles for both parent and child traffic rules.

Restrictions and guidelines

A level-4 rule cannot act as a parent rule.

You can specify a parent traffic rule only when creating a traffic rule. You cannot add or modify a parent traffic rule for an existing traffic rule.

Procedure

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. (Optional.) Enable bandwidth management for traffic flows of the IP layer and upper layers.
   all-traffic-control enable

   By default, bandwidth management is performed only for traffic flows of Layer 4 and upper layers.

   Use this feature when there is a large number of IP traffic flows in the network.
4. Create a traffic rule and enter traffic rule view.
   rule [*rule-id*] **name** *rule-name* [**parent** *parent-rule-name*]

   You can specify a traffic rule as the parent traffic rule for multiple child traffic rules.

Configuring match criteria for the traffic rule

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Enter traffic rule view.

Choose one option as needed:

- **rule** *rule-id*
- **rule** [*rule-id*] **name** *rule-name* [**parent** *parent-rule-name*]

4. Configure a security zone as a match criterion.

- Configure a destination security zone as a match criterion.

  destination-zone *destination-zone-name*
- Configure a source security zone as a match criterion.

  source-zone *source-zone-name*

By default, no security zone is used as a match criterion.

5. Configure an IP address object group as a match criterion.

- Configure a destination IP address object group as a match criterion.

  destination-address address-set *object-group-name*
- Configure a source IP address object group as a match criterion.

  source-address address-set *object-group-name*

By default, no IP address object group is used as a match criterion.

6. Configure a service object group as a match criterion.

service *object-group-name*

By default, no service object group is used as a match criterion.

7. Configure an application or application group as a match criterion.

application { **app** *application-name* | **app-group** *application-group-name* }

By default, no application or application group is used as a match criterion.

8. Configure a user or user group as a match criterion.

- Configure a user as a match criterion.

  user *user-name* [**domain** *domain-name*]
- Configure a user group as a match criterion.

  user-group *user-group-name* [**domain** *domain-name*]

By default, no user or user group is used as a match criterion.

9. Configure a DSCP priority as a match criterion.

dscp *dscp-value*

By default, no DSCP priority is used as a match criterion.

10. Configure an IPv6 packet attribute as a match criterion.

- Configure the flow label attribute as a match criterion

  ipv6 flow-label { **nonzero** | **zero** }

  By default, the flow label attribute is not used as a match criterion.
- Configure the extension header attribute as a match criterion

  ipv6 extension-header { **authentication** | **destination** | **encapsulating** | **fragment** | **hop-by-hop** | **routing** }

  By default, the extension header attribute is not used as a match criterion.

Support for this command depends on the device model. For more information, see the command reference.

Support for this command depends on the device model. For more information, see the command reference.

11. Configure a terminal or terminal group as a match criterion.

- Configure a terminal as a match criterion.

terminal *terminal-name*

By default, no terminal is used as a match criterion.

- Configure a terminal group as a match criterion.

  terminal-group *group-name*

  By default, no terminal group is used as a match criterion.

Specifying an action for the traffic rule

About this task

If a packet matches a traffic rule, the device performs the action specified in the traffic rule on the packet.

Restrictions and guidelines

When you specify traffic profiles for parent and child traffic rules, make sure the following conditions are met:

- The maximum bandwidth for a child traffic rule must be smaller than or equal to that for the parent traffic rule.
- The guaranteed bandwidth for a child traffic rule must be smaller than or equal to that for the parent traffic rule.
- The traffic profiles cannot be the same for the child and parent traffic rules.

Procedure

1. Enter system view.

   system-view

2. Enter traffic policy view.

   traffic-policy

3. Enter traffic rule view.

   Choose one option as needed:

   - **rule** *rule-id*
   - **rule** [*rule-id*] **name** *rule-name* [**parent** *parent-rule-name*]

4. Specify an action for the traffic rule.

   action { **deny** | **none** | **qos profile** *profile-name* }

   The default action is **none**, which allows matching packets to pass through without bandwidth management.

Specifying a time range for the traffic rule

1. Enter system view.

   system-view

2. Enter traffic policy view.

   traffic-policy

3. Enter traffic rule view.

   Choose one option as needed:

   - **rule** *rule-id*
   - **rule** [*rule-id*] **name** *rule-name* [**parent** *parent-rule-name*]

4. Specify a time range during which the traffic rule is in effect.

time-range *time-range-name*

By default, a traffic rule is in effect at any time.

Managing and maintaining a traffic rule

Copying a traffic rule

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Copy a traffic rule.
   rule copy *rule-name new-rule-name*

Renaming a traffic rule

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Rename a traffic rule.
   rule rename *old-rule-name new-rule-name*

Moving a traffic rule

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Move a traffic rule to a new position.
   rule move *rule-name1* { **after** *rule-name2* | **before** [*rule-name2*] }

Disabling a traffic rule

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Enter traffic rule view.
   Choose one option as needed:
   - **rule** *rule-id*
   - **rule** [*rule-id*] **name** *rule-name* [**parent** *parent-rule-name*]
4. Disable the traffic rule.
   disable

By default, a traffic rule is enabled.

Enabling bandwidth management statistics collection

About this task

This feature can collect the following statistics:

- Traffic statistics.
- Connection limit statistics.
- Rule-hit statistics.

Restrictions and guidelines

This feature affects device performance. As a best practice, enable this feature only if you need to view statistics.

Procedure

1. Enter system view.
   system-view
2. Enter traffic policy view.
   traffic-policy
3. Enable bandwidth management statistics collection.
   - Enable traffic statistics collection.
     statistics bandwidth enable
     By default, traffic statistics collection is disabled.
   - Enable connection limit statistics collection.
     statistics connection-limit enable
     By default, connection limit statistics collection is disabled.
   - Enable rule-hit statistics collection.
     statistics rule-hit enable
     By default, rule-hit statistics collection is disabled.

Display and maintenance commands for bandwidth management

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display traffic statistics for traffic rules.	**display traffic-policy statistics bandwidth** { **downstream** | **total** | **upstream** } { **per-ip** { **ipv4** [*ipv4-address*] | **ipv6** [*ipv6-address*] } **rule** *rule-name* | **per-rule** [**name** *rule-name*] | **per-user** [**user** *user-name*] **rule** *rule-name* } [**slot** *slot-number*]
Display connection limit statistics.	**display traffic-policy statistics connection-limit** { **per-ip** { **ipv4** [*ipv4-address*]

Task	Command
	| **ipv6** [*ipv6-address*] } **rule** *rule-name* | **per-rule** [**name** *rule-name*] | **per-user** [**user** *user-name*] **rule** *rule-name* } } [**slot** *slot-number*]
Display rule-hit statistics.	**display traffic-policy statistics rule-hit** [**rule** *rule-name*] [**slot** *slot-number*]
Clear traffic statistics for traffic rules.	**reset traffic-policy statistics bandwidth** { **downstream** | **total** | **upstream** } { **per-ip** { **ipv4** [*ipv4-address*] | **ipv6** [*ipv6-address*] } **rule** *rule-name* | **per-rule** [**name** *rule-name*] | **per-user** [**user** *user-name*] **rule** *rule-name* } [**slot** *slot-number*]
Clear connection limit statistics.	**reset traffic-policy statistics connection-limit** { **per-ip** { **ipv4** [*ipv4-address*] | **ipv6** [*ipv6-address*] } **rule** *rule-name* | **per-rule** [**name** *rule-name*] | **per-user** [**user** *user-name*] **rule** *rule-name* } } [**slot** *slot-number*]
Clear rule-hit statistics.	**reset traffic-policy statistics rule-hit** [**rule** *rule-name*] [**slot** *slot-number*]

Bandwidth management configuration examples

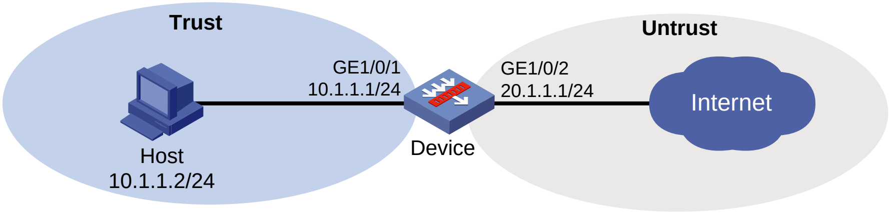
Example: Configuring a single traffic profile

Network configuration

As shown in Figure 2, configure bandwidth management on the device to meet the following requirements:

- The maximum bandwidth is limited to 30720 kbps for both upstream and downstream iQiYiPPS application traffic of the host in the intranet.
- The guaranteed bandwidth is 30720 kbps for both upstream and downstream FTP traffic of the host .
- The bandwidth of the interface to the Internet is limited to 102400 kbps.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
```

```
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. **Configure settings for routing.**

   This example configures static routes, and the next hop in the routes is 20.1.1.2.

   ```
   [Device] ip route-static 3.1.1.2 24 20.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow the host to access the Internet.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-1-trust-untrust] source-zone trust
   [Device-security-policy-ip-1-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.2
   [Device-security-policy-ip-1-trust-untrust] action pass
   [Device-security-policy-ip-1-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure traffic profiles:

   # Create a traffic profile named **aiqiyi**, and enter traffic profile view.

   ```
   [Device] traffic-policy
   [Device-traffic-policy] profile name aiqiyi
   ```

   # Set the maximum bandwidth to 30720 kbps for both upstream and downstream traffic.

   ```
   [Device-traffic-policy-profile-aiqiyi] bandwidth upstream maximum 30720
   [Device-traffic-policy-profile-aiqiyi] bandwidth downstream maximum 30720
   [Device-traffic-policy-profile-aiqiyi] quit
   ```

   # Create a traffic profile named **profileftp**, and enter traffic profile view.

   ```
   [Device-traffic-policy] profile name profileftp
   ```

   # Set the guaranteed bandwidth to 30720 kbps for both upstream and downstream traffic.

   ```
   [Device-traffic-policy-profile-profileftp] bandwidth upstream guaranteed 30720
   [Device-traffic-policy-profile-profileftp] bandwidth downstream guaranteed 30720
   [Device-traffic-policy-profile-profileftp] quit
   [Device-traffic-policy] quit
   ```

6. Set the expected bandwidth to 102400 kbps for interface GigabitEthernet 1/0/2.

   ```
   [Device] interface gigabitethernet 1/0/2
   [Device-GigabitEthernet1/0/2] bandwidth 102400
   [Device-GigabitEthernet1/0/2] quit
   ```

7. Configure traffic rules:

   # Enter traffic policy view.

   ```
   [Device] traffic-policy
   ```

   # Create a traffic rule named **aiqiyi**, and enter traffic rule view.

   ```
   [Device-traffic-policy] rule name aiqiyi
   ```

   # Configure the predefined application iQiYiPPS as a match criterion.

```
[Device-traffic-policy-rule-1-aiqiyi] application app iQiYiPPS
```

# Specify traffic profile **aiqiyi** for traffic rule **aiqiyi**.

```
[Device-traffic-policy-rule-1-aiqiyi] action qos profile aiqiyi
[Device-traffic-policy-rule-1-aiqiyi] quit
```

# Create a traffic rule named **ruleftp**, and enter traffic rule view.

```
[Device-traffic-policy] rule name ruleftp
```

# Configure the predefined application FTP as a match criterion.

```
[Device-traffic-policy-rule-2-ruleftp] application app ftp
```

# Specify traffic profile **profileftp** for traffic rule **ruleftp**.

```
[Device-traffic-policy-rule-2-ruleftp] action qos profile profileftp
[Device-traffic-policy-rule-2-ruleftp] quit
[Device-traffic-policy] quit
```

Verifying the configuration

# Verify that the iQiYiPPS application traffic rate cannot exceed 30720 kbps and the FTP traffic rate can reach a minimum of 30720 kbps when the total traffic rate on GigabitEthernet 1/0/2 reaches 102400 kbps. (Details not shown.)

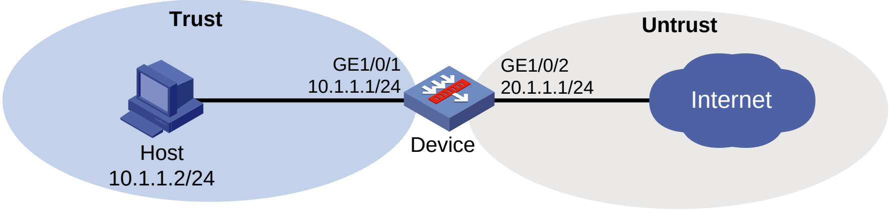
Example: Configuring parent/child traffic profiles

Network configuration

As shown in Figure 3, configure bandwidth management on the device to meet the following requirements:

- The maximum bandwidth is limited to 30720 kbps for both upstream and downstream iQiYiPPS application traffic of the host in the intranet.
- The guaranteed bandwidth is 30720 kbps for both upstream and downstream FTP traffic of the host .
- The total traffic rate of the host is limited to 40960 kbps.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 20.1.1.2.

   ```
   [Device] ip route-static 3.1.1.2 24 20.1.1.2
   ```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

# Configure a rule named **trust-untrust** to allow the host to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure traffic profiles:

# Create a traffic profile named **profile**, and enter traffic profile view.

```
[Device] traffic-policy
[Device-traffic-policy] profile name profile
```

# Set the maximum bandwidth to 40960 kbps for both upstream and downstream traffic.

```
[Device-traffic-policy-profile-profile] bandwidth upstream maximum 40960
[Device-traffic-policy-profile-profile] bandwidth downstream maximum 40960
[Device-traffic-policy-profile-profile] quit
```

# Create a traffic profile named **aiqiyi**, and enter traffic profile view.

```
[Device-traffic-policy] profile name aiqiyi
```

# Set the maximum bandwidth to 30720 kbps for both upstream and downstream traffic.

```
[Device-traffic-policy-profile-aiqiyi] bandwidth upstream maximum 30720
[Device-traffic-policy-profile-aiqiyi] bandwidth downstream maximum 30720
[Device-traffic-policy-profile-aiqiyi] quit
```

# Create a traffic profile named **profileftp**, and enter traffic profile view.

```
[Device-traffic-policy] profile name profileftp
```

# Set the guaranteed bandwidth to 30720 kbps for both upstream and downstream traffic.

```
[Device-traffic-policy-profile-profileftp] bandwidth upstream guaranteed 30720
[Device-traffic-policy-profile-profileftp] bandwidth downstream guaranteed 30720
[Device-traffic-policy-profile-profileftp] quit
```

6. Configure traffic rules:

# Create a traffic rule named **rule**, and enter traffic rule view.

```
[Device-traffic-policy] rule name rule
```

# Specify traffic profile **profile** for traffic rule **rule**.

```
[Device-traffic-policy-rule-1-rule] action qos profile profile
[Device-traffic-policy-rule-1-rule] quit
```

# Create a traffic rule named **aiqiyi**, enter traffic rule view, and specify traffic rule **rule** as its parent rule.

```
[Device-traffic-policy] rule name aiqiyi parent rule
```

# Configure the predefined application iQiYiPPS as a match criterion.

```
[Device-traffic-policy-rule-2-aiqiyi] application app iQiYiPPS
```

# Specify traffic profile **aiqiyi** for traffic rule **aiqiyi**.

```
[Device-traffic-policy-rule-2-aiqiyi] action qos profile aiqiyi
[Device-traffic-policy-rule-2-aiqiyi] quit
```

# Create a traffic rule named **ruleftp**, enter traffic rule view, and specify traffic rule **rule** as its parent rule.

```
[Device-traffic-policy] rule name ruleftp parent rule
```

# Configure the predefined application FTP as a match criterion.

```
[Device-traffic-policy-rule-3-ruleftp] application app ftp
```

# Specify traffic profile **profileftp** for traffic rule **ruleftp**.

```
[Device-traffic-policy-rule-3-ruleftp] action qos profile profileftp
[Device-traffic-policy-rule-3-ruleftp] quit
[Device-traffic-policy] quit
```

Verifying the configuration

# Verify that the total traffic rate of the host is limited to 40960 kbps, and that the iQiYiPPS application traffic rate is limited to 30720 kbps. When congestion occurs, FTP traffic is not affected. (Details not shown.)

Example: Configuring a user-based traffic profile

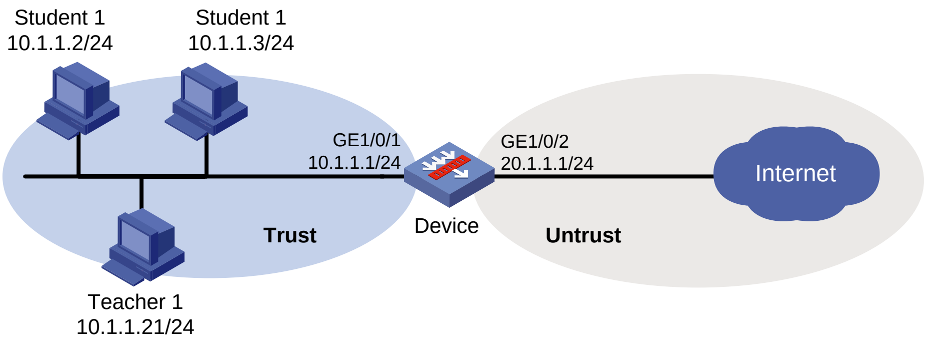
Network configuration

As shown in Figure 4, an intranet has two user groups: a teacher group with two teachers and a student group with five students.

Configure per-user bandwidth management on the device to meet the following requirements:

- The IP addresses of students **student1** through **student5** are 10.1.1.2/24 through 10.1.1.6/24, respectively. The IP addresses of teachers **teacher1** and **teacher2** are 10.1.1.21/24 and 10.1.1.22/24, respectively.
- The bandwidth is limited to 10000 kbps for each teacher in both the upstream and downstream directions, and the bandwidth is limited to 2000 kbps for each student in both the upstream and downstream directions.
- The connection count limit is limited to 10000 for each teacher, and the connection count limit is also limited to 10000 for each student.
- Teachers have higher priority over students to access the Internet.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

This example configures static routes, and the next hop in the routes is 20.1.1.2.

```
[Device] ip route-static 3.1.1.2 24 20.1.1.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

# Configure a rule named **trust-untrust** to allow the host to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.2
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.3
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.4
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.5
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.6
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.21
[Device-security-policy-ip-1-trust-untrust] source-ip-host 10.1.1.22
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure local users:

# Create a network access user named **student1**.

```
[Device] local-user student1 class network
```

# Set the password to **student** in plaintext form for the user.

```
[Device-luser-network-student1] password simple student
```

# Specify the service types as IKE, Portal, and SSL VPN for the user.

```
[Device-luser-network-student1] service-type ike
[Device-luser-network-student1] service-type portal
[Device-luser-network-student1] service-type sslvpn
[Device-luser-network-student1] quit
```

# Create four network access users named **student2**, **student3**, **student4**, and **student5**, and set the password to **student** in plaintext form for each of these users. (Details not shown.)

# Create two network access users named **teacher1** and **teacher2**, and set the password to **teacher** in plaintext form for each of these users. (Details not shown.)

# Specify the service types as IKE, Portal, and SSL VPN for each of the six users. (Details not shown.)

# Create a user group named **student**, and assign users **student1** through **student5** to the user group as identity users.

```
[Device] user-group student
[Device-ugroup-student] identity-member user student1
[Device-ugroup-student] identity-member user student2
[Device-ugroup-student] identity-member user student3
[Device-ugroup-student] identity-member user student4
[Device-ugroup-student] identity-member user student5
[Device-ugroup-student] quit
```

# Create a user group named **teacher**, and assign users **teacher 1** and **teacher2** to the user group as identity users.

```
[Device] user-group teacher
[Device-ugroup-teacher] identity-member user teacher1
[Device-ugroup-teacher] identity-member user teacher2
[Device-ugroup-teacher] quit
```

# Configure static identity users.

```
[Device] user-identity static-user student1 bind ipv4 10.1.1.2
[Device] user-identity static-user student2 bind ipv4 10.1.1.3
[Device] user-identity static-user student3 bind ipv4 10.1.1.4
[Device] user-identity static-user student4 bind ipv4 10.1.1.5
[Device] user-identity static-user student5 bind ipv4 10.1.1.6
[Device] user-identity static-user teacher1 bind ipv4 10.1.1.21
[Device] user-identity static-user teacher2 bind ipv4 10.1.1.22
```

# Enable the user identification feature.

```
[Device] user-identity enable
```

6. Configure traffic profiles:

# Create a traffic profile named **profile-teacher**, and enter traffic profile view.

```
[Device] traffic-policy
[Device-traffic-policy] profile name profile-teacher
```

# Set the per-user maximum bandwidth to 10000 kbps for both upstream and downstream traffic.

```
[Device-traffic-policy-profile-profile-teacher] bandwidth upstream maximum per-user
10000
[Device-traffic-policy-profile-profile-teacher] bandwidth downstream maximum
per-user 10000
```

# Set the per-user connection count limit to 10000.

```
[Device-traffic-policy-profile-profile-teacher] connection-limit count per-user
10000
```

# Set the traffic priority to 2.

```
[Device-traffic-policy-profile-profile-teacher] traffic-priority 2
```

# Create a traffic profile named **profile-student**, and enter traffic profile view.

```
[Device-traffic-policy] profile name profile-student
```

# Set the per-user maximum bandwidth to 2000 kbps for both upstream and downstream traffic.

```
[Device-traffic-policy-profile-profile-student] bandwidth upstream maximum per-user
2000
[Device-traffic-policy-profile-profile-student] bandwidth downstream maximum
per-user 2000
```

# Set the per-user connection count limit to 10000.

```
[Device-traffic-policy-profile-profile-student] connection-limit count per-user
10000
```

# Set the traffic priority to 1 (lowest).

```
[Device-traffic-policy-profile-profile-student] traffic-priority 1
```

7. Configure traffic rules:

# Create a traffic rule named **rule-teacher**, and enter traffic rule view.

```
[Device-traffic-policy] rule name rule-teacher
```

# Configure user group **teacher** as a match criterion, and specify traffic profile **profile-teacher** for traffic rule **rule-teacher**.

```
[Device-traffic-policy-rule-1-rule-teacher] user-group teacher
[Device-traffic-policy-rule-1-rule-teacher] action qos profile profile-teacher
[Device-traffic-policy-rule-1-rule-teacher] quit
```

# Create a traffic rule named **rule-student**, and enter traffic rule view.

```
[Device-traffic-policy] rule name rule-student
```

# Configure user group **student** as a match criterion, and specify traffic profile **profile-student** for traffic rule **rule-student**.

```
[Device-traffic-policy-rule-2-rule-student] user-group student
[Device-traffic-policy-rule-2-rule-student] action qos profile profile-student
[Device-traffic-policy-rule-2-rule-student] quit
[Device-traffic-policy] quit
```

Verifying the configuration

# Verify that the bandwidth is limited to 10000 kbps for each teacher, and that the bandwidth is limited to 2000 kbps for each student.

# Verify that the per-user connection count is limited to 10000 for both students and teachers.

Contents

Configuring application audit and management

About application audit and management

Application audit and management audits and records Internet access behaviors of users by identifying behaviors (for example, login and message sending in IM applications) and behavior objects (for example, account information for IM login).

NOTE:

This feature parses personal information from user packets and must be used for legitimate purposes.

Application audit and management policy

You can configure match criteria, audit rules, and actions in an application audit and management policy to audit matching packets.

Policy types

Application audit and management policies have the following types:

- **Audit policy**—Audits packets that meet match criteria in the policy.
- **Audit-free policy**—Does not audit packets that meet match criteria in the policy.
- **Deny policy**—Drops packets that meet match criteria in the policy.

Match criteria

Multiple match criteria can be configured in an application audit and management policy.

The following match criteria are available:

- Source and destination security zones.
- Source and destination IP addresses.
- Services.
- Users/user groups.
- Applications.

One match criterion can contain multiple match values. For example, you can configure multiple source security zones for a source security zone match criterion.

Audit rule

Audit rules can be configured for an audit policy to perform more granular control on user behaviors and to generate audit logs.

The following rule match modes are available:

- **in-order**—The device compares packets with audit rules in ascending order of rule ID. When a packet matches a rule, the device stops the match process and performs the action defined in the rule.
- **all**—The device compares packets with audit rules in ascending order of rule ID.
  - If a packet matches a rule with the permit action, all subsequent rules continue to be matched.

The device takes the action with higher priority on matching packets. The deny action has higher priority than the permit action.

  - If a packet matches a rule with the deny action, the device stops the match process and performs the deny action.

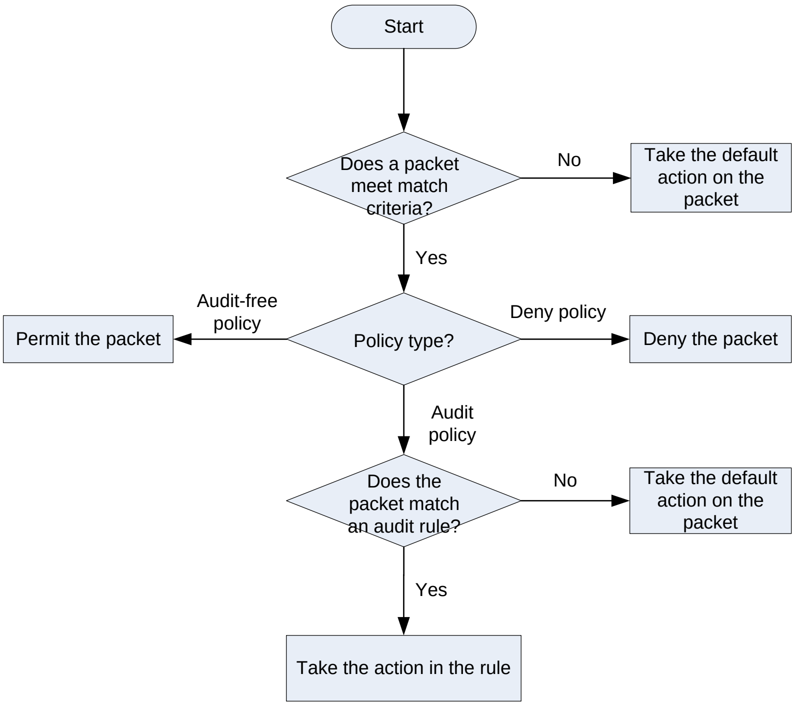
Audit log

The device can generate and output audit logs for packets that match an audit rule. The audit logs can be output as common logs or output by using the fast log output feature.

Application audit and management workflow

Figure 1 shows the application audit and management workflow.

Figure 1 Application audit and management workflow


Start
Does a packet meet match criteria?
No
Take the default action on the packet
Yes
Audit-free policy
Permit the packet
Policy type?
Deny policy
Deny the packet
Audit policy
Does the packet match an audit rule?
No
Take the default action on the packet
Yes
Take the action in the rule

The application audit and management workflow is as follows:

1. The device matches the packet against the match criteria in an application audit and management policy.

   The packet meets a match criterion if it matches any of its match values. A packet does not match a match criterion if it matches none of its match values.

2. If the packet meets all match criteria in the policy (for the user and user group criteria, only one criterion needs to be matched), the packet matches the policy. Otherwise, the packet does not match the policy and continues to be matched by the next policy. If the packet does not match any policy, the device takes the policy's default action on the packet.

3. If the packet matches a policy, it is processed according to the policy type.
   - If the policy is an audit-free policy, the packet is allowed to pass.
   - If the policy is a deny policy, the packet is denied.
   - If the policy is an audit policy, the packet is matched against the audit rules in the policy.

4. The device processes the packet as follows:
   - If a packet matches all items in an audit rule, the action in the audit rule is taken on the packet.

  - If a packet matches only the specified application or application category in an audit rule, the packet is allowed to pass through.
  - If a packet does not match the specified application or application category in an audit rule, the default action for audit rules is taken on the packet.

Restrictions and guidelines: Application audit and management configuration

As a best practice to audit packets more accurately, observe the depth-first principle when creating policies. Always create a policy with a smaller audit scope before a policy with a larger audit scope.

Prerequisites for application audit and management

Before configuring application audit and management, complete the following tasks:

- Update the APR signature library to the latest version (see APR configuration in *Security Configuration Guide*).
- Configure time ranges (see time range configuration in *ACL and QoS Configuration Guide*).
- Configure IP address object groups and service object groups (see object group configuration in *Security Configuration Guide*).
- Configure applications (see APR configuration in *Security Configuration Guide*).
- Configure users and user groups (see user identification configuration in *Security Configuration Guide*).
- Configure security zones (see security zone configuration in *Security Configuration Guide*).

Application audit and management tasks at a glance

To configure application audit and management, perform the following tasks:

1. Creating an application audit and management policy
2. Configuring match criteria for the policy
3. (Optional.) Specifying a time range for the policy
4. Configuring an audit rule for the policy
5. Configuring a keyword group
6. (Optional.) Managing and maintaining an application audit and management policy
7. Activating policy and rule settings of all DPI service modules

Creating an application audit and management policy

1. Enter system view.

   ```
   system-view
   ```

2. Enter application audit and management view.

uapp-control

3. Create an application audit and management policy and enter its view.

   policy name *policy-name* { **audit** | **deny** | **noaudit** }

4. Configure the default action for the policy.

   policy default-action { **deny** | **permit** }

   By default, the default action for a policy is **permit**.

Configuring match criteria for the policy

1. Enter system view.

   system-view

2. Enter application audit and management view.

   uapp-control

3. Enter application audit and management policy view.

   policy name *policy-name* [**audit** | **deny** | **noaudit**]

4. Configure a security zone as a match criterion.
   - Configure a source security zone as a match criterion.

     source-zone *source-zone-name*

   - Configure a destination security zone as a match criterion.

     destination-zone *destination-zone-name*

   By default, no security zone is used as a match criterion.

5. Configure an IP address object group as a match criterion.
   - Configure a source IP address object group as a match criterion.

     source-address { **ipv4** | **ipv6** } *object-group-name*

   - Configure a destination IP address object group as a match criterion.

     destination-address { **ipv4** | **ipv6** } *object-group-name*

   By default, no IP address object group is used as a match criterion.

6. Configure a service object group as a match criterion.

   service *service-name*

   By default, no service object group is used as a match criterion.

7. Configure a user or user group as a match criterion.
   - Configure a user as a match criterion.

     user *user-name* [**domain** *domain-name*]

   - Configure a user group as a match criterion.

     user-group *user-group-name* [**domain** *domain-name*]

   By default, no user or user group is used as a match criterion.

8. Configure an application or application group as a match criterion.

   application { **app** *application-name* | **app-group** *application-group-name* }

   By default, no application or application group is used as a match criterion.

   The application and application group match criteria can be configured only in audit-free policies and deny policies.

Specifying a time range for the policy

1. Enter system view.

   `system-view`

2. Enter application audit and management view.

   `uapp-control`

3. Enter application audit and management policy view.

   `policy name` *`policy-name`* { **`audit`** | **`deny`** | **`noaudit`** }

4. Specify a time range during which the policy is in effect.

   `time-range` *`time-range-name`*

   By default, an application audit and management policy is in effect at any time.

Configuring an audit rule for the policy

About this task

An audit rule provices the following functions:

- **General auditing**—Performs granular control on user behaviors.
- **Email protection**—Detects incoming emails, counts emails based on recipients, and protects recipients from attacks. Specifically, you can configure the following functions:
  - **Limit email sending**—Prevents users from sending emails to users of a different domain. For example, the user at user1@abc.com cannot receive emails from the user at user2@123.com.
  - **Prevent email bombing**—Protects recipients from being overwhelmed by large numbers of emails from the same sender during a short period of time.

You can configure audit rules only for an audit policy.

If you specify the **`audit-logging`** keyword for an audit rule, the following rules apply:

- The device sends audit log messages to the information center as common logs by default. With the information center, you can set log message filtering and output rules, including output destinations. The information center can output audit logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect. To view audit logs stored on the device, use the **`display logbuffer`** command. Make sure you do not disable log output to the log buffer, which is enabled by default. For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.
- If you also configure the **`customlog format dpi audit`** command, the device outputs audit logs by using the fast log output feature. For more information about fast log output, see fast log output configuration in *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

For WeChat and QQ, specific messages and voice calls cannot be blocked..

Procedure

1. Enter system view.

   `system-view`

2. Enter application audit and management view.

   `uapp-control`

3. Enter application audit and management policy view of the audit type.

policy name *policy-name* [**audit**]

4. Configure an audit rule for the policy.

   rule *rule-id* { **app** *app-name* | **app-category** *app-category-name* | **any** } **behavior** { *behavior-name* | **any** } **bhcontent** { *bhcontent-name* | **any** } { **keyword** { **equal** | **exclude** | **include** | **unequal** } { *keyword-group-name* | **any** } | **integer** { **equal** | **greater** | **greater-equal** | **less** | **less-equal** | **unequal** } { *number* } } **action** { **deny** | **permit** } [**audit-logging**]

   rule *rule-id* { **email-bomb-defense** [**interval** *interval* **max-number** *email-number*] | **email-send-restriction** } * **action** { **deny** | **permit** } [**audit-logging**]

   By default, a policy does not have audit rules.

5. Configure the match mode for audit rules in the policy.

   rule match-method { **all** | **in-order** }

   By default, the match mode for audit rules is **in-order**.

6. Configure the default action for audit rules in the policy.

   rule default-action { **deny** | **permit** }

   By default, the default action for audit rules is **permit**.

Configuring a keyword group

About this task

A keyword group can be used by an audit rule to match more specific information.

Procedure

1. Enter system view.

   system-view

2. Enter application audit and management view.

   uapp-control

3. Create a keyword group and enter its view.

   keyword-group name *keyword-group-name*

4. (Optional.) Configure a description for the keyword group.

   description *text*

   By default, a keyword group does not have a description.

5. Add a keyword to the keyword group.

   keyword *keyword-value*

   By default, a keyword group does not contain keywords.

Managing and maintaining an application audit and management policy

Copying an application audit and management policy

1. Enter system view.

   system-view

2. Enter application audit and management view.

uapp-control

3. Copy an application audit and management policy.

   policy copy *policy-name new-policy-name*

Renaming an application audit and management policy

1. Enter system view.

   system-view

2. Enter application audit and management view.

   uapp-control

3. Rename an application audit and management policy.

   policy rename *old-policy-name new-policy-name*

Moving an application audit and management policy

1. Enter system view.

   system-view

2. Enter application audit and management view.

   uapp-control

3. Move an application audit and management policy.

   policy move *policy-name1* { **after** *policy-name2* | **before** [*policy-name2*] }

Disabling an application audit and management policy

1. Enter system view.

   system-view

2. Enter application audit and management view.

   uapp-control

3. Enter application audit and management policy view.

   policy name *policy-name*

4. Disable the application audit and management policy.

   disable

   By default, an application audit and management policy is enabled.

Activating policy and rule settings of all DPI service modules

About this task

After a policy or rule of a DPI service module is created, modified, or deleted, you must perform this task for the configuration to take effect.

For more information about this task, see DPI engine configuration in *DPI Configuration Guide*.

Restrictions and guidelines

This task will interrupt DPI service processing. To reduce the impact on DPI services, perform this task after you complete policy and rule settings of all DPI service modules.

Procedure

1. Enter system view.

   `system-view`

2. Activate policy and rule settings of all DPI service modules.

   `inspect activate`

   By default, policy and rule settings of all DPI service modules are deactivated.

Application audit and management configuration examples

Example: Configuring account login audit

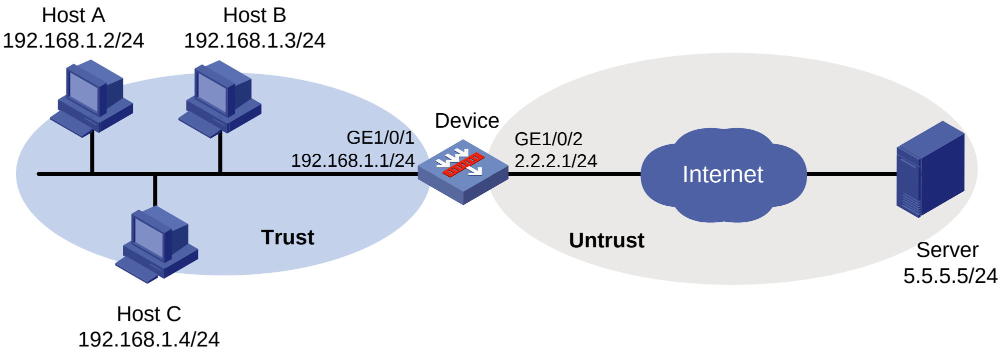
Network configuration

As shown in Figure 2, all departments of a company access the Internet through the device. The working hours of the company are 8:00:00 through 18:00:00 from Monday to Friday.

Configure an application audit and management policy on the device to meet the following requirements:

- Permit login from all QQ accounts during working hours.
- Generate audit logs.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[Device] ip route-static 5.5.5.5 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow the host to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.2
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.3
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.4
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure a time range named **work** to cover 8:00:00 through 18:00:00 from Monday to Friday.

```
[Device] time-range work 08:00 to 18:00 working-day
```

6. Update the APR signature library to the latest version (in this example, 1.0.106). (Details not shown.)

7. Configure an application audit and management policy:

   # Enter application audit and management view.

```
[Device] uapp-control
```

   # Create an audit policy named **audit-qq** and enter its view.

```
[Device-uapp-control] policy name audit-qq audit
```

   # Configure source security zone **Trust** as a match criterion for audit policy **audit-qq**.

```
[Device-uapp-control-policy-audit-qq] source-zone trust
```

   # Configure destination security zone **Untrust** as a match criterion for audit policy **audit-qq**.

```
[Device-uapp-control-policy-audit-qq] destination-zone untrust
```

   # Specify time range **work** for audit policy **audit-qq**.

```
[Device-uapp-control-policy-audit-qq] time-range work
```

   # Configure an audit rule to permit login from all QQ accounts and generate audit logs.

```
[Device-uapp-control-policy-audit-qq] rule 1 app QQ behavior Login bhcontent any
keyword equal any action permit audit-logging
[Device-uapp-control-policy-audit-qq] quit
[Device-uapp-control] quit
```

   # Activate the configuration.

```
[Device] inspect activate
```

Verifying the configuration

When QQ accounts attempt to access the Internet, the device permits the login requests and generates audit log messages.

Example: Configuring sensitive information audit

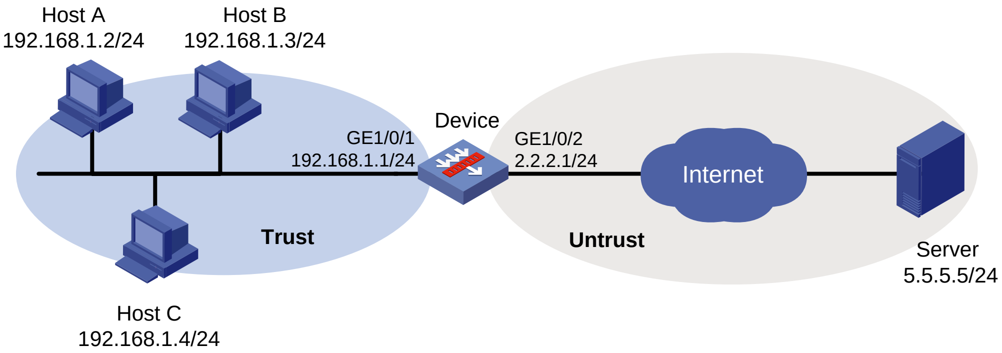
Network configuration

As shown in Figure 3, all departments of a company access the Internet through the device. The working hours of the company are 8:00:00 through 18:00:00 from Monday to Friday.

Configure an application audit and management policy on the device to meet the following requirements:

- Deny search requests for Bing that include keyword **confidential** or **terrorist attack**.
- Generate audit logs.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hop in the routes is 2.2.2.2.

```
[Device] ip route-static 5.5.5.5 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to allow the host to access the Internet.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.2
```

```
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.3
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.4
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Update the APR signature library to the latest version (in this example, 1.0.106). (Details not shown.)
6. Configure an application audit and management policy:

   # Enter application audit and management view.

   ```
   [Device] uapp-control
   ```

   # Configure a keyword group named **keyword-bing**.

   ```
   [Device-uapp-control] keyword-group name keyword-bing
   ```

   # Add keywords **confidential** and **terrorist attack** to keyword group **keyword-bing**.

   ```
   [Device-uapp-control-keyword-group-keyword-bing] keyword confidential
   [Device-uapp-control-keyword-group-keyword-bing] keyword terrorist attack
   [Device-uapp-control-keyword-group-keyword-bing] quit
   ```

   # Create an audit policy named **audit-bing** and enter its view.

   ```
   [Device-uapp-control] policy name audit-bing audit
   ```

   # Configure source security zone **Trust** as a match criterion for audit policy **audit-bing**.

   ```
   [Device-uapp-control-policy-audit-bing] source-zone trust
   ```

   # Configure destination security zone **Untrust** as a match criterion for audit policy **audit-bing**.

   ```
   [Device-uapp-control-policy-audit-bing] destination-zone untrust
   ```

   # Configure an audit rule to deny search requests for Bing that include keyword **confidential** or **terrorist attack**, generating audit logs.

   ```
   [Device-uapp-control-policy-audit-bing] rule 2 app Bing behavior Search bhcontent
   Keyword keyword include keyword-bing action deny audit-logging
   [Device-uapp-control-policy-audit-bing] quit
   [Device-uapp-control] quit
   ```

   # Activate the configuration.

   ```
   [Device] inspect activate
   ```

Verifying the configuration

When a user searches for information that includes keyword **confidential** or **terrorist attack** by using Bing, the device denies the search request and generates a log message.

Contents

Configuring NetShare control

About NetShare control

NetShare control uses the NetShare control policy to identify and control network sharing behaviors.

The network sharing behavior is the behavior of multiple terminals using the same IP address for network access through NAT or proxy. If an IP address is detected to be used as the source IP address in packets sent by multiple terminals, the IP address is a shared IP address. NetShare control monitors the number of terminals sharing the IP address and takes the NetShare control action if the number of terminals sharing the IP address exceeds the limit.

NetShare detection methods

NetShare control uses the following methods to detect network sharing behaviors:

- **APR-based detection**—The device analyzes the application layer information of packets based on the Application Recognition (APR)-based packet analysis to detect NetShare behaviors of terminals. For more information about APR, see APR configuration in *Security Configuration Guide*.
- **IPID trail tracking**—The device tracks the values of the IPID fields in packets to detect NetShare behaviors.

NetShare control policy

The device uses the NetShare control policy to detect NetShare behaviors.

A NetShare control policy defines the following attributes:

- Filtering criteria.
- Detection methods.
- Maximum number of terminals allowed to share an IP address.
- Actions on terminals that exceed the NetShare control upper limit.

If the number of terminals sharing the IP address exceeds the limit, NetShare control takes the NetShare control actions specified in the policy.

NetShare control mechanism

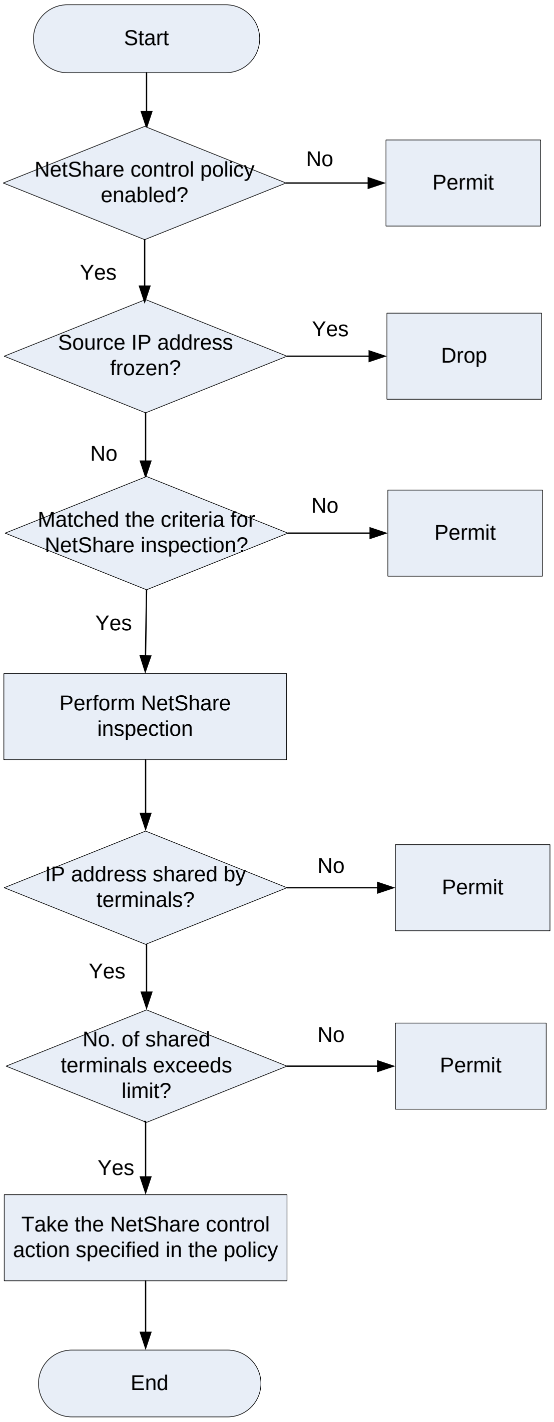
As shown in Figure 1, the NetShare control module processes a packet as follows:

1. Determines if the NetShare policy is enabled.
   - If the policy is disabled, NetShare control permits the packet to pass through.
   - If the policy is enabled, NetShare control proceeds to step 2.
2. Determines if the source IP address of the packet is frozen:
   - If yes, NetShare control drops the packet.
   - If not, NetShare control proceeds to step 3.
3. Compares the packet attributes with the NetShare inspection criteria in the NetShare control policy to determine if the packet matches the policy.
   - If the packet does not match the policy, NetShare control permits the packet to pass through.
   - If the packet matches the policy, NetShare control proceeds to step 4.

4. Determines if the source IP address of the packet is shared by multiple terminals:

- If not, NetShare control permits the packet to pass through.
- If yes, NetShare control further determines whether the number of terminals sharing the IP address exceeds the limit:
  - If the limit is exceeded, NetShare control takes the NetShare control action specified in the policy.
  - If the limit is not exceeded, NetShare control permits the packet to pass through.

Figure 1 NetShare control mechanism


Start
NetShare control policy enabled?
No
Permit
Yes
Source IP address frozen?
Yes
Drop
No
Matched the criteria for NetShare inspection?
No
Permit
Yes
Perform NetShare inspection
IP address shared by terminals?
No
Permit
Yes
No. of shared terminals exceeds limit?
No
Permit
Yes
Take the NetShare control action specified in the policy
End

NetShare control tasks at a glance

To configure NetShare control, perform the following tasks:

1. Creating a NetShare control policy
2. Configuring NetShare inspection filtering criteria
3. Configuring a NetShare detection method
    - Enabling APR-based detection
    - Enabling IPID trail tracking
4. Setting the maximum number of terminals sharing an IP address
5. Setting the NetShare control action
6. Activating NetShare control policy settings
7. (Optional.) Disabling the NetShare control policy
8. (Optional.) Manually freezing and unfreezing a shared IP address

Prerequisites for NetShare control

Before you configure NetShare control, you must perform the following tasks:

- Upgrade the APR signature library on the device to the most recent version.
- Configure IP address object groups. For information about the configuration procedure, see object group configuration in *Security Configuration Guide*.
- Configure users and user groups. For information about the configuration procedures, see user identification configuration in *Security Configuration Guide*.
- Configure security zones. For information about the configuration procedure, see security zone configuration in *Security Configuration Guide*.

Creating a NetShare control policy

Restrictions and guidelines

The device supports only one NetShare control policy.

Procedure

1. Enter system view.
   `system-view`
2. Enter NetShare control configuration view.
   `netshare-control`
3. Create a NetShare control policy and enter its view.
   `policy name` *`policy-name`*
4. (Optional.) Configure a description for the NetShare control policy.
   `description` *`string`*
   By default, a NetShare control policy does not have a description.

Configuring NetShare inspection filtering criteria

About this task

In the NetShare control policy, you can configure multiple criteria of different criterion types to filter the packets to be analyzed for NetShare inspection. A packet must match a minimum of one criterion in each configured criterion type to be inspected by the NetShare control module.

The following filtering criterion types are supported:

- Source IP address.
- Destination IP address.
- Source security zone.
- Destination security zone.
- User, including username- and user group-based filtering criteria.

Procedure

1. Enter system view.

   `system-view`

2. Enter NetShare control configuration view.

   `netshare-control`

3. Enter NetShare control policy view.

   `policy name` *`policy-name`*

4. Configure source and destination security zone criteria:
   - Configure a source security zone criterion.

     `source-address` { **`ipv4`** | **`ipv6`** } *`object-group-name`*

     By default, the NetShare control policy does not contain any source security zone criterion.
   - Configure a destination security zone criterion.

     `destination-address` { **`ipv4`** | **`ipv6`** } *`object-group-name`*

     By default, the NetShare control policy does not contain any destination security zone criterion.
5. Configure source and destination address criteria:
   - Configure a source address criterion.

     `source-address` { **`ipv4`** | **`ipv6`** } *`object-group-name`*

     By default, the NetShare control policy does not contain any source address criterion.
   - Configure a destination address criterion:

     `destination-address` { **`ipv4`** | **`ipv6`** } *`object-group-name`*

     By default, the NetShare control policy does not contain any destination address criterion.
6. Configure user and user group criteria:
   - Configure a user criterion.

     `user` *`username`* [**`domain`** *`domain-name`*]

     By default, the NetShare control policy does not contain any user criterion.
   - Configure a user group criterion.

     `user-group` *`user-group-name`* [**`domain`** *`domain-name`*]

     By default, the NetShare control policy does not contain any user group criterion.

Configuring a NetShare detection method

Enabling APR-based detection

About this task

This feature supports detecting only a limited set of applications in the APR signature library.

Restrictions and guidelines

You can enable both APR-based detection and IPID trail tracking to detect NetShare behaviors.

APR-based NetShare detection uses the APR signature library to inspect only specific applications, such as QQ and WeChat. If an application is encrypted, APR-based NetShare detection cannot inspect it. As a best practice, enable APR-based detection only when explicitly required, because the detection might degrade the device performance.

Procedure

1. Enter system view.
   `system-view`
2. Enter NetShare control configuration view.
   `netshare-control`
3. Enter NetShare control policy view.
   `policy name` *`policy-name`*
4. Enable APR-based detection in the NetShare control policy.
   `application-inspect enable`
   By default, APR-based detection is enabled.

Enabling IPID trail tracking

About this task

By default, the device uses only the APR-based detection method to detect NetShare behaviors. APR-based NetShare detection applies only to a limited set of applications in the APR signature library. To meet the NetShare control requirements of various application scenarios, you can enable the IPID trail tracking method so the device can use both detection methods for NetShare behavior detection.

IPID trail tracking tracks the values of the IPID fields in packets to detect NetShare behaviors.

Restrictions and guidelines

You can enable both APR-based detection and IPID trail tracking to detect NetShare behaviors.

IPID trail tracking might degrade the device performance. Enable it only when explicitly required.

IPID trail tracking supports detecting the terminals that are running the Windows system, and detecting packets in which values of the IPID fields change regularly. Mobile terminals are not supported.

IPID trail tracking supports detecting IPv4 packets.

Procedure

1. Enter system view.
   `system-view`
2. Enter NetShare control configuration view.

netshare-control

3. Enter NetShare control policy view.

   policy name *policy-name*

4. Enable IPID trail tracking in the NetShare control policy.

   ipid-trail enable

   By default, IPID trail tracking is disabled in the NetShare control policy.

Setting the maximum number of terminals sharing an IP address

About this task

If the number of terminals sharing an IP address exceeds the limit, the device will take the NetShare control action set in the NetShare control policy.

Procedure

1. Enter system view.

   system-view

2. Enter NetShare control configuration view.

   netshare-control

3. Enter NetShare control policy view.

   policy name *policy-name*

4. Set the maximum number of terminals that can share an IP address.

   per-ip-shared max-terminals *number*

   By default, the number of terminals that can share an IP address is not limited.

Setting the NetShare control action

About this task

The NetShare control action is taken when the number of terminals sharing an IP address exceeds the limit.

The following NetShare control actions are supported:

- **Freeze**—Freezes the shared IP address for the specified freezing time. All packets sourced from the frozen IP address will be dropped.
- **Permit**—Permits the packets sourced from the IP address to pass through.
- **Logging**—Logs the NetShare control event.

Restrictions and guidelines

The **logging** keyword enables the NetShare control module to log NetStream control events and send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output NetShare control logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view NetShare control logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter NetShare control configuration view.

   netshare-control

3. Enter NetShare control policy view.

   policy name *policy-name*

4. Set the NetShare control action.

   action { **freeze** *freeze-time* | **permit** } [**logging**]

   By default, the NetShare control policy uses the **permit** action.

Activating NetShare control policy settings

About this task

After you create or delete a NetShare control policy, perform this task to activate the configuration.

Restrictions and guidelines

This task can cause temporary outage for all DPI services. As a best practice, perform the task after all DPI service policy and rule settings are complete.

For more information about activating DPI service module configuration, see DPI engine configuration in *DPI Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Activate NetShare control policy settings.

   inspect activate

   By default, NetShare control policy creation and deletion do not take effect.

Disabling the NetShare control policy

About this task

If the NetShare control feature is not required on the network, disable the NetShare control policy.

Restrictions and guidelines

The device supports only one NetShare control policy. After you disable the NetShare control policy, the NetShare control feature becomes invalid.

Procedure

1. Enter system view.

   system-view

2. Enter NetShare control configuration view.

   netshare-control

3. Enter NetShare control policy view.

   policy name *policy-name*

4. Disable the NetShare control policy.

   disable

   By default, a NetShare control policy is enabled.

Manually freezing and unfreezing a shared IP address

About this task

You can manually unfreeze a frozen IP address or freeze a shared IP address that is not in frozen state.

Procedure

1. Enter system view.

   system-view

2. Enter NetShare control configuration view.

   netshare-control

3. Manually freeze a shared IP address.

   freeze { **ipv4** | **ipv6** } **ip-address** [**vpn-instance** *vpn-instance-name*] **time** *freeze-time*

4. Manually unfreeze a frozen IP address.

   unfreeze { **ipv4** | **ipv6** } **ip-address** [**vpn-instance** *vpn-instance-name*]

Display and maintenance commands for NetShare control

Execute **display** commands in any view.

Task	Command
Display NetShare control information about shared IP addresses.	**display netshare-control** [{ **ipv4** | **ipv6** } **ip-address** | **status** { **frozen** | **unfrozen** }] [**slot** *slot-number*]

NetShare control configuration examples

Example: Configuring NetShare control

Network configuration

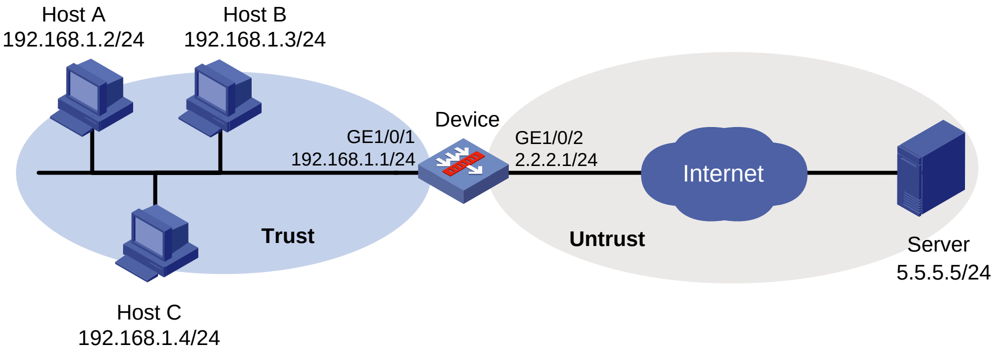
As shown in Figure 2, the device connects to the LAN and Internet through security zones **Trust** and **Untrust**, respectively.

Configure NetShare control on the device to meet the following requirements:

- Monitor the packets sent by the hosts on the LAN to the Internet for network sharing behavior inspection.

- If an IP address is detected to be shared by more than one host for Internet access, NetShare control will freeze the IP address for 1 hour and logs the event.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 192.168.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route to reach the server, and the next hop in the route is 2.2.2.2.

```
[Device] ip route-static 5.5.5.0 24 2.2.2.2
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
```

4. Enter IPv4 security policy view and create a security policy rule named **trust-untrust** to permit the packets from security zone **Trust** to security zone **Untrust**.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.2
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.3
[Device-security-policy-ip-1-trust-untrust] source-ip-host 192.168.1.4
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Create a NetShare control policy named **a**. Configure NetShare control to freeze an IP address for 1 hour if the number of endpoints sharing the IP address exceeds 1 and to log the event.

```
[Device] netshare-control
```

```
[Device-netshare-control] policy name a
[Device-netshare-control-policy-a] source-zone trust
[Device-netshare-control-policy-a] destination-zone untrust
[Device-netshare-control-policy-a] per-ip-shared max-terminals 1
[Device-netshare-control-policy-a] action freeze 60 logging
[Device-netshare-control-policy-a] quit
[Device-netshare-control] quit
```

6. Activate the NetShare control policy settings.

```
[Device] inspect activate
```

Verifying the configuration

# Verify that if a host on the LAN accesses the Internet by using a shared IP address through a proxy, the device can detect the network sharing behavior. In addition, the device will freeze the shared IP address for 1 hour and log the event. (Details not shown.)

NSFOCUS Firewall Series

NF Load Balancing Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for load balancing features.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

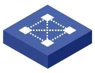
GUI conventions

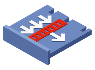
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Load balancing overview

Load balancing (LB) is a cluster technology that distributes services among multiple network devices or links.

Advantages of load balancing

Load balancing has the following advantages:

- **High performance**—Improves overall system performance by distributing services to multiple devices or links.
- **Scalability**—Meets increasing service requirements without compromising service quality by easily adding devices or links.
- **High availability**—Improves overall availability by using backup devices or links.
- **Manageability**—Simplifies configuration and maintenance by centralizing management on the load balancing device.
- **Transparency**—Preserves the transparency of the network topology for end users. Adding or removing devices or links does not affect services.

Load balancing types

LB includes the following types:

Link load balancing—Link load balancing applies to a network environment where there are multiple carrier links to implement dynamic link selection. This enhances link utilization. Link load balancing supports IPv4 and IPv6, but does not support IPv4-to-IPv6 packet translation. Link load balancing is classified into the following types based on the direction of connection requests:

- **Outbound link load balancing**—Load balances traffic among the links from the internal network to the external network.
- **Inbound link load balancing**—Load balances traffic among the links from the external network to the internal network.
- **Transparent DNS proxy**—Load balances DNS requests among the links from the internal network to the external network.

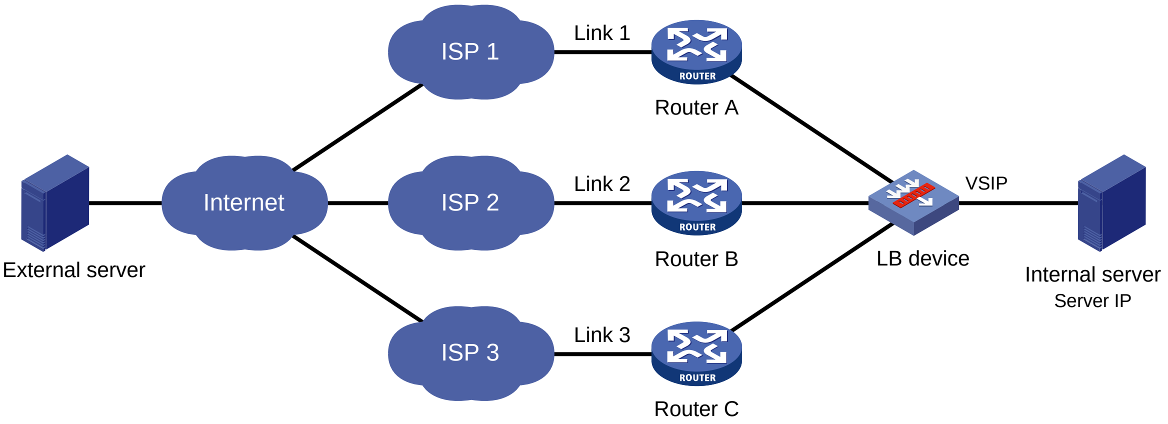
Configuring outbound link load balancing

About outbound link load balancing

Outbound link load balancing load balances traffic among the links from the internal network to the external network.

Typical network diagram

Figure 1 Network diagram



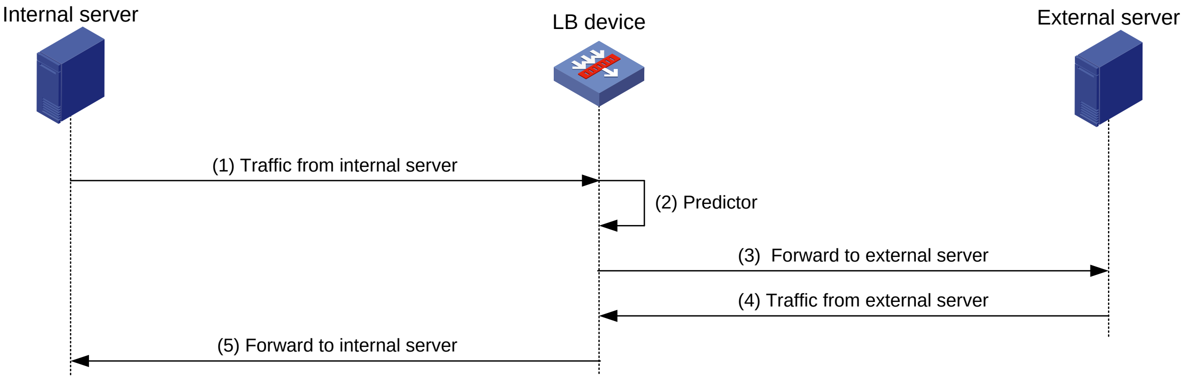
As shown in Figure 1, outbound link load balancing contains the following elements:

- **LB device**—Distributes outbound traffic among multiple links.
- **Link**—Physical links provided by ISPs.
- **VSIP**—Virtual service IP address of the cluster, which identifies the destination network for packets from the internal network.
- **Server IP**—IP address of a server.

Workflow

Figure 2 shows the outbound link load balancing workflow.

Figure 2 Outbound link load balancing workflow


Internal server
LB device
External server
(1) Traffic from internal server
(2) Predictor
(3) Forward to external server
(4) Traffic from external server
(5) Forward to internal server

The workflow for outbound link load balancing is as follows:

1. The LB device receives traffic from the internal server.
2. The LB device selects the optimal link based on the LB policy, sticky method, proximity algorithm, and scheduling algorithm (typically the bandwidth algorithm or maximum bandwidth algorithm) in turn.
3. The LB device forwards the traffic to the external server through the optimal link.
4. The LB device receives traffic from the external server.
5. The LB device forwards the traffic to the internal server.

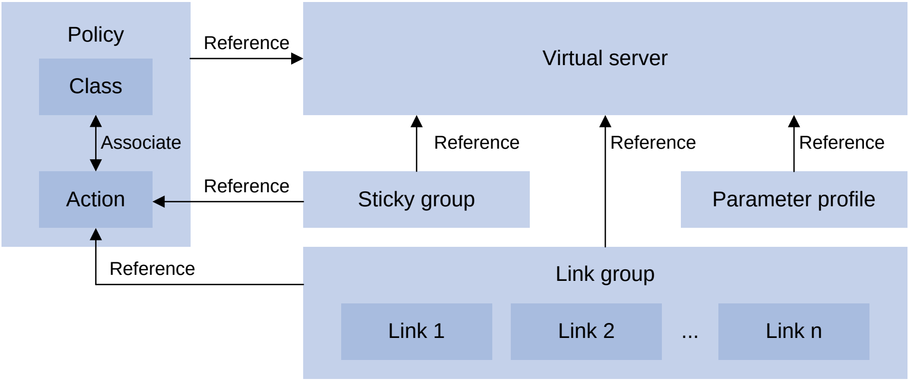
Outbound link load balancing tasks at a glance

Relationship between configuration items

Figure 3 shows the relationship between the following configuration items:

- **Link group**—A collection of links that contain similar functions. A link group can be referenced by a virtual server or an LB action.
- **Link**—Physical links provided by ISPs.
- **Virtual server**—A virtual service provided by the LB device to determine whether to perform load balancing for packets received on the LB device. Only the packets that match a virtual server are load balanced.
- **LB class**—Classifies packets to implement load balancing based on packet type.
- **LB action**—Drops, forwards, or modifies packets.
- **LB policy**—Associates an LB class with an LB action. An LB policy can be referenced by a virtual server.
- **Sticky group**—Uses a sticky method to distribute similar sessions to the same link. A sticky group can be referenced by a virtual server or an LB action.
- **Parameter profile**—Defines advanced parameters to process packets. A parameter profile can be referenced by a virtual server.

Figure 3 Relationship between the main configuration items



Tasks at a glance

To configure outbound link load balancing, perform the following tasks:

1. Configuring a link group
2. Configuring a link

3. Configuring a virtual server
4. (Optional.) Configuring an LB policy
    a. Configuring an LB class
    b. Configuring an LB action
    c. Configuring an LB policy
5. (Optional.) Configuring a sticky group
6. (Optional.) Configuring a parameter profile
7. (Optional.) Configuring ISP information
8. (Optional.) Setting the aging time for DNS cache entries
9. (Optional.) Configuring the ALG feature
10. (Optional.) Performing a load balancing test
11. (Optional.) Configuring SNMP notifications and logging for load balancing
    - Enabling SNMP notifications
    - Enabling load balancing logging

Configuring a link group

You can add links that contain similar functions to a link group to facilitate management.

Link group tasks at a glance

To configure a link group, perform the following tasks:

1. Creating a link group
2. (Optional.) Adding and configuring a link group member
3. Configuring a scheduling algorithm for a link group
4. Setting the availability criteria
5. Disabling NAT
6. (Optional.) Configuring SNAT
7. (Optional.) Enabling the slow online feature
8. (Optional.) Configuring health monitoring
9. (Optional.) Specifying a fault processing method
10. (Optional.) Configuring the proximity feature

Creating a link group

1. Enter system view.
    system-view
2. Create a link group and enter link group view.
    loadbalance link-group *link-group-name*
3. (Optional.) Configure a description for the link group.
    description *text*

    By default, no description is configured for a link group.

Adding and configuring a link group member

About this task

Perform this task to create a link group member or add an existing link as a link group member in link group view. You can also specify a link group for a link in link view to achieve the same purpose (see "Creating a link and specifying a link group").

After adding a link group member, you can configure the following parameters and features for the link in the link group:

- Weight.
- Priority.
- Connection limits.
- Health monitoring.
- Slow offline.

The member-based scheduling algorithm selects the best link based on these configurations.

Adding a link group member

1. Enter system view.
   `system-view`
2. Enter link group view.
   `loadbalance link-group` *`link-group-name`*
3. Create and add a link group member and enter link group member view.
   `link` *`link-name`*
   If the link already exists, the command adds the existing link as a link group member.
4. (Optional.) Configure a description for the link group member.
   `description` *`text`*
   By default, no description is configured for the link group member.

Setting the weight and priority of the link group member

1. Enter system view.
   `system-view`
2. Enter link group view.
   `loadbalance link-group` *`link-group-name`*
3. Enter link group member view.
   `link` *`link-name`*
4. Set the weight of the link group member.
   `weight` *`weight-value`*
   The default setting is 100.
5. Set the priority of the link group member.
   `priority` *`priority`*
   The default setting is 4.

Setting the connection limits of the link group member

1. Enter system view.
   `system-view`
2. Enter link group view.
   `loadbalance link-group` *`link-group-name`*

3. Enter link group member view.

   `link` *`link-name`*

4. Set the connection rate of the link group member.

   `rate-limit connection` *`connection-number`*

   The default setting is 0 (the connection rate is not limited).

5. Set the maximum number of connections allowed for the link group member.

   `connection-limit max` *`max-number`*

   The default setting is 0 (the maximum number of connections is not limited).

Configuring health monitoring for the link group member

1. Enter system view.

   `system-view`

2. Enter link group view.

   `loadbalance link-group` *`link-group-name`*

3. Enter link group member view.

   `link` *`link-name`*

4. Specify a health monitoring method for the link group member.

   `probe` *`template-name`*

   By default, no health monitoring method is specified for the link group member.

   You can specify an NQA template for health monitoring. For information about NQA templates, see NQA configuration in *Network Management and Monitoring Configuration Guide*.

5. Specify the health monitoring success criteria for the link group member.

   `success-criteria` { **`all`** | **`at-least`** *`min-number`* }

   By default, health monitoring succeeds only when all the specified health monitoring methods succeed.

Enabling the slow offline feature for the link group member

1. Enter system view.

   `system-view`

2. Enter link group view.

   `loadbalance link-group` *`link-group-name`*

3. Enter link group member view.

   `link` *`link-name`*

4. Enable the slow offline feature for the link group member.

   `slow-shutdown enable`

   By default, the slow offline feature is disabled.

5. Shut down the link group member.

   `shutdown`

   By default, the link group member is activated.

Configuring a scheduling algorithm for a link group

About this task

Perform this task to specify a scheduling algorithm for a link group and specify the number of links to participate in scheduling. The LB device calculates the links to process user requests based on the specified scheduling algorithm.

The device provides the following scheduling algorithms for a link group:

- **Weighted least connection algorithm (link-based)**—Always assigns user requests to the link with the fewest number of weighted active connections (the total number of active connections in all link groups divided by weight). The weight value used in this algorithm is configured in real server view.
- **Weighted least connection algorithm (link group member-based)**—Always assigns user requests to the link with the fewest number of weighted active connections (the total number of active connections in the specified link group divided by weight). The weight value used in this algorithm is configured in link group member view.
- **Random algorithm**—Randomly assigns user requests to links.
- **Round robin algorithm**—Assigns user requests to links based on the weights of links. A higher weight indicates more user requests will be assigned.
- **Bandwidth algorithm**—Distributes user requests to links according to the weights and remaining bandwidth of links.
- **Maximum bandwidth algorithm**—Distributes user requests always to an idle link that has the largest remaining bandwidth.
- **Source IP address hash algorithm**—Hashes the source IP address of user requests and distributes user requests to different links according to the hash values.
- **Source IP address and port hash algorithm**—Hashes the source IP address and port number of user requests and distributes user requests to different links according to the hash values.
- **Destination IP address hash algorithm**—Hashes the destination IP address of user requests and distributes user requests to different links according to the hash values.

Procedure

1. Enter system view.

   system-view

2. Enter link group view.

   loadbalance link-group *link-group-name*

3. Specify a scheduling algorithm for the link group.

   - Specify a link-based scheduling algorithm.

     predictor { **least-connection** | { **bandwidth** | **max-bandwidth** } [**inbound** | **outbound**] }

   - Specify a link group member-based scheduling algorithm.

     predictor hash address { **destination** | **source** | **source-ip-port** } [**mask** *mask-length*] [**prefix** *prefix-length*]

     predictor { **least-connection member** | **random** | **round-robin** }

   By default, the scheduling algorithm for a link group is weighted round robin.

4. Specify the number of links to participate in scheduling.

   selected-link min *min-number* **max** *max-number*

   By default, the links with the highest priority participate in scheduling.

Setting the availability criteria

About this task

Perform this task to set the criteria (lower percentage and higher percentage) to determine whether a link group is available. This helps implement traffic switchover between the master and backup link groups.

- When the number of available links to the total number of links in the master link group is smaller than the lower percentage, traffic is switched to the backup link group.
- When the number of available links to the total number of links in the master link group is greater than the upper percentage, traffic is switched back to the master link group.

Procedure

1. Enter system view.

   system-view
2. Enter link group view.

   loadbalance link-group *link-group-name*
3. Set the criteria to determine whether the link group is available.

   activate lower *lower-percentage* **upper** *upper-percentage*

   By default, when a minimum of one link is available, the link group is available.

Disabling NAT

Restrictions and guidelines

Typically, outbound link load balancing networking requires disabling NAT for a link group.

Procedure

1. Enter system view.

   system-view
2. Enter link group view.

   loadbalance link-group *link-group-name*
3. Disable NAT for the link group.

   transparent enable

   By default, NAT is enabled for a link group.

Configuring SNAT

About this task

After a link group references the SNAT address pool, the LB device replaces the source address of the packets it receives with an SNAT address before forwarding the packets.

Restrictions and guidelines

An SNAT address pool can have multiple address ranges. Each address range can have a maximum of 256 IPv4 addresses or 65536 IPv6 addresses. No overlapping IPv4 or IPv6 addresses are allowed in the same SNAT address pool or different SNAT address pools.

As a best practice, do not use SNAT because its application scope is limited for outbound link load balancing.

Procedure

1. Enter system view.

   system-view
2. Create a SNAT address pool and enter SNAT address pool view.

   loadbalance snat-pool *pool-name*
3. (Optional.) Configure a description for the SNAT address pool.

   description *text*

By default, no description is configured for a SNAT address pool.

4. Specify an address range for the SNAT address pool.

   IPv4:

   ip range start *start-ipv4-address* **end** *end-ipv4-address*

   IPv6:

   ipv6 range start *start-ipv6-address* **end** *end-ipv6-address*

   By default, a SNAT address pool does not contain address ranges.

5. (Optional.) Specify a VPN instance for the SNAT address pool.

   vpn-instance *vpn-instance-name*

   By default, a SNAT address pool belongs to the public network.

   Use this command to separate overlapping SNAT address pools.

6. Return to system view.

   quit

7. Enter link group view.

   loadbalance link-group *link-group-name*

8. Specify the SNAT address pool to be referenced by the link group.

   snat-pool *pool-name*

   By default, no SNAT address pool is referenced by a link group.

Enabling the slow online feature

About this task

Links newly added to a link group might be unable to immediately process large numbers of services assigned by the LB device. To resolve this issue, enable the slow online feature for the link group. The feature uses the standby timer and ramp-up timer. When the links are brought online, the LB device does not assign any services to the links until the standby timer expires.

When the standby timer expires, the ramp-up timer starts. During the ramp-up time, the LB device increases the service amount according to the processing capability of the links, until the ramp-up timer expires.

Procedure

1. Enter system view.

   system-view

2. Enter link group view.

   loadbalance link-group *link-group-name*

3. Enable the slow online feature for the link group.

   slow-online [**standby-time** *standby-time* **ramp-up-time** *ramp-up-time*]

   By default, the slow online feature is disabled for a link group.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect the availability of links.

Restrictions and guidelines

The health monitoring configuration in link view takes precedence over the configuration in link group view.

You can specify an NQA template for health monitoring. For information about NQA templates, see NQA configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter link group view.

   `loadbalance link-group` *`link-group-name`*

3. Specify a health monitoring method for the link group.

   `probe` *`template-name`*

   By default, no health monitoring method is specified for a link group.

4. Specify the health monitoring success criteria for the link group.

   `success-criteria` { **`all`** | **`at-least`** *`min-number`* }

   By default, health monitoring succeeds only when all the specified health monitoring methods succeed.

Specifying a fault processing method

About this task

Perform this task to specify one of the following fault processing methods for a link group:

- **Keep**—Does not actively terminate the connection with the failed link. Keeping or terminating the connection depends on the timeout mechanism of the protocol.
- **Reschedule**—Redirects the connection to another available link in the link group.
- **Reset**—Terminates the connection with the failed link by sending RST packets (for TCP packets) or ICMP unreachable packets (for other types of packets).

Procedure

1. Enter system view.

   `system-view`

2. Enter link group view.

   `loadbalance link-group` *`link-group-name`*

3. Specify a fault processing method for the link group.

   `fail-action` { **`keep`** | **`reschedule`** | **`reset`** }

   By default, the fault processing method is **`keep`**. All available connections are kept.

Configuring the proximity feature

About this task

The proximity feature performs link detection to select the optimal link to a destination. If no proximity information for a destination is available, the load balancing module selects a link based on the scheduling algorithm. It then performs proximity detection to generate proximity entries for forwarding subsequent traffic.

You can specify an NQA template or load-balancing probe template to perform link detection. The device generates proximity entries according to the detection results and proximity parameter settings. For information about NQA templates, see NQA configuration in *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

To configure the proximity feature, first configure proximity parameters in proximity view, and then enable the proximity feature in link group view.

Configuring an LB probe template

1. Enter system view.

   system-view
2. Create an LB probe template and enter LB probe template view.

   loadbalance probe-template icmp *template-name*
3. Set the probe interval.

   frequency *interval*

   The default setting is 300 seconds.
4. Set the timeout time for probe responses.

   timeout *timeout-value*

   The default setting is 3 seconds.

Configuring the proximity probe method

1. Enter system view.

   system-view
2. Enter proximity view.

   loadbalance proximity [**vpn-instance** *vpn-instance-name*]
3. Specify the proximity probe method for packets.

   match [*match-id*] { **tcp** } { **lb-probe** *lb-template* | **probe** *nqa-template* }

   By default, no proximity probe method is specified.
4. Specify the default proximity probe method.

   match default { **lb-probe** *lb-template* | **probe** *nqa-template* }

   By default, the default proximity probe method is not specified.

Configuring proximity parameters

1. Enter system view.

   system-view
2. Enter proximity view.

   loadbalance proximity [**vpn-instance** *vpn-instance-name*]
3. Set the mask length for IPv4 proximity entries.

   ip mask { *mask-length* | *mask* }

   By default, the mask length for IPv4 proximity entries is 24.
4. Set the prefix length for IPv6 proximity entries.

   ipv6 prefix *prefix-length*

   By default, the prefix length for IPv6 proximity entries is 96.
5. Set the network delay weight for proximity calculation.

   rtt weight *rtt-weight*

   By default, the network delay weight for proximity calculation is 100.
6. Set the TTL weight for proximity calculation.

   ttl weight *ttl-weight*

   By default, the TTL weight for proximity calculation is 100.
7. Set the bandwidth weight for proximity calculation.

bandwidth { **inbound** | **outbound** } **weight** *bandwidth-weight*

By default, the inbound or outbound bandwidth weight for proximity calculation is 100.

8. Set the cost weight for proximity calculation.

   cost weight *cost-weight*

   By default, the cost weight for proximity calculation is 100.

9. Set the aging timer for proximity entries.

   timeout *timeout-value*

   By default, the aging timer for proximity entries is 60 seconds.

10. Set the maximum number of proximity entries.

    max-number *number*

    By default, the maximum number of proximity entries is 65535.

Enabling the proximity feature

1. Enter system view.

   system-view

2. Enter link group view.

   loadbalance link-group *link-group-name*

3. Enable the proximity feature.

   proximity enable

   By default, the proximity feature is disabled for a link group.

Configuring a link

A link is a physical link provided by an ISP. A link can belong to multiple link groups. A link group can have multiple links.

Restrictions and guidelines

In a network where hot backup cooperates with VRRP, as a best practice, make sure the IP address of the link-attached output interface is on the same network segment as the virtual IP address of the VRRP group. For more information about hot backup and VRRP association, see hot backup configuration in *High Availability Configuration Guide*.

Link tasks at a glance

To configure a link, perform the following tasks:

1. Creating a link and specifying a link group
2. Specifying a next hop IP address or an outgoing interface

   Choose one of the following tasks:

   - Specifying an outbound next hop for a link
   - Specifying an outgoing interface for a link
3. Setting a weight and priority
4. (Optional.) Configuring the bandwidth and connection parameters
5. (Optional.) Configuring health monitoring
6. (Optional.) Enabling the slow offline feature
7. (Optional.) Setting the link cost for proximity calculation

8. (Optional.) Setting the bandwidth ratio and maximum expected bandwidth
9. (Optional.) Disabling VPN instance inheritance for a link

Creating a link and specifying a link group

1. Enter system view.

   system-view
2. Create a link and enter link view.

   loadbalance link *link-name*

   By default, no links exist.
3. (Optional.) Configure a description for the link.

   description *text*

   By default, no description is configured for a link.
4. Specify a link group for the link.

   link-group *link-group-name*

   By default, a link does not belong to any link group.

Specifying an outbound next hop for a link

1. Enter system view.

   system-view
2. Enter link view.

   loadbalance link *link-name*
3. Specify an outbound next hop for the link.

   IPv4:

   router ip *ipv4-address*

   IPv6:

   router ipv6 *ipv6-address*

   By default, a link does not have an outbound next hop.

Specifying an outgoing interface for a link

About this task

In scenarios where IP addresses are obtained through PPPoE, an LB device can dynamically obtain the outbound next hop IP address through the specified outgoing interface.

Procedure

1. Enter system view.

   system-view
2. Enter link view.

   loadbalance link *link-name*
3. Specify an outgoing interface for the link.

   router interface *interface-type interface-number*

   By default, no outgoing interface is specified for a link.

Setting a weight and priority

About this task

Perform this task to configure a weight for the weighted round robin and weighted least connection algorithms of a link, and the scheduling priority in the link group for the server.

Procedure

1. Enter system view.

   `system-view`

2. Enter link view.

   `loadbalance link` *`link-name`*

3. Set a weight for the link.

   `weight` *`weight-value`*

   By default, the weight of a link is 100.

4. Set a priority for the link.

   `priority` *`priority`*

   By default, the priority of a link is 4.

Configuring the bandwidth and connection parameters

1. Enter system view.

   `system-view`

2. Enter link view.

   `loadbalance link` *`link-name`*

3. Set the maximum bandwidth for the link.

   `rate-limit bandwidth` [**`inbound`** | **`outbound`**] *`bandwidth-value`* **`kbps`**

   By default, the maximum bandwidth, inbound bandwidth, and outbound bandwidth are 0 for a link. The bandwidths are not limited.

4. Set the maximum number of connections for the link.

   `connection-limit max` *`max-number`*

   By default, the maximum number of connections is 0 for a link. The number is not limited.

5. Set the maximum number of connections per second for the link.

   `rate-limit connection` *`connection-number`*

   By default, the maximum number of connections per second is 0 for a link. The number is not limited.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect the availability of a link.

Restrictions and guidelines

The health monitoring configuration in link view takes precedence over the configuration in link group view.

Procedure

1. Enter system view.

system-view

2. Enter link view.

   loadbalance link *link-name*

3. Specify a health monitoring method for the link.

   probe *template-name*

   By default, no health monitoring method is specified for a link.

4. Specify the health monitoring success criteria for the link.

   success-criteria { **all** | **at-least** *min-number* }

   By default, the health monitoring succeeds only when all the specified health monitoring methods succeed.

Enabling the slow offline feature

About this task

The **shutdown** command immediately terminates existing connections of a link. The slow offline feature ages out the connections, and does not establish new connections.

Restrictions and guidelines

To enable the slow offline feature for a link, you must execute the **slow-shutdown enable** command and then the **shutdown** command. If you execute the **shutdown** command and then the **slow-shutdown enable** command, the slow offline feature does not take effect and the link is shut down.

Procedure

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Enable the slow offline feature for the link.

   slow-shutdown enable

   By default, the slow offline feature is disabled.

4. Shut down the link.

   shutdown

   By default, the link is activated.

Setting the link cost for proximity calculation

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Set the link cost for proximity calculation.

   cost *cost-value*

   By default, the link cost for proximity calculation is 0.

Setting the bandwidth ratio and maximum expected bandwidth

About this task

When the traffic exceeds the maximum expected bandwidth multiplied by the bandwidth ratio of a link, new traffic (traffic that does not match any sticky entries) is not distributed to the link. When the traffic drops below the maximum expected bandwidth multiplied by the bandwidth recovery ratio of the link, the link participates in scheduling again.

In addition to being used for link protection, the maximum expected bandwidth is used for remaining bandwidth calculation in the bandwidth algorithm, maximum bandwidth algorithm, and dynamic proximity algorithm.

Procedure

1. Enter system view.

`system-view`

2. Enter link view.

`loadbalance link` *`link-name`*

3. Set the bandwidth ratio.

`bandwidth` [**`inbound`** | **`outbound`**] **`busy-rate`** *`busy-rate-number`* [**`recovery`** *`recovery-rate-number`*]

By default, the total bandwidth ratio is 70.

4. Set the maximum expected bandwidth.

`max-bandwidth` [**`inbound`** | **`outbound`**] *`bandwidth-value`* **`kbps`**

By default, the maximum expected bandwidth, maximum uplink expected bandwidth, and maximum downlink expected bandwidth are 0. The bandwidths are not limited.

Disabling VPN instance inheritance for a link

About this task

When VPN instance inheritance is enabled, a link without a VPN instance specified inherits the VPN instance of the virtual server. When VPN instance inheritance is disabled, a link without a VPN instance specified belongs to the public network.

Procedure

1. Enter system view.

`system-view`

2. Enter link view.

`loadbalance link` *`link-name`*

3. Disable VPN instance inheritance for the link.

`inherit vpn-instance disable`

By default, VPN instance inheritance is enabled for a link.

Configuring a virtual server

A virtual server is a virtual service provided by the LB device to determine whether to perform load balancing for packets received on the LB device. Only the packets that match a virtual server are load balanced.

Restrictions and guidelines

Outbound link load balancing supports only the link-IP virtual server.

Virtual server tasks at a glance

To configure a virtual server, perform the following tasks:

1. Creating a virtual server
2. Specifying the VSIP and port number
3. (Optional.) Specifying a VPN instance
4. Configuring a packet processing policy

   Choose the following tasks as needed:

   - Specifying link groups
   - Specifying an LB policy
5. (Optional.) Specifying a parameter profile
6. (Optional.) Configuring the bandwidth and connection parameters
7. (Optional.) Enabling the link protection feature
8. (Optional.) Enabling bandwidth statistics collection by interfaces
9. (Optional.) Specifying a DPI application profile
10. (Optional.) Configuring hot backup
11. Enabling a virtual server

Creating a virtual server

1. Enter system view.

   system-view
2. Create a link-IP virtual server and enter virtual server view.

   virtual-server *virtual-server-name* **type link-ip**
3. (Optional.) Configure a description for the virtual server.

   description *text*

   By default, no description is configured for the virtual server.

Specifying the VSIP and port number

1. Enter system view.

   system-view
2. Enter link-IP virtual server view.

   virtual-server *virtual-server-name*
3. Specify the VSIP for the virtual server.

   IPv4:

   virtual ip address *ipv4-address* [*mask-length* | *mask*]

   IPv6:

   virtual ipv6 address *ipv6-address* [*prefix-length*]

   By default, no IPv4 or IPv6 address is specified for a virtual server.
4. Specify the port number for the virtual server.

port *port-number*

By default, the port number is 0 (meaning any port number) for a link-IP virtual server.

Specifying a VPN instance

1. Enter system view.
   system-view
2. Enter virtual server view.
   virtual-server *virtual-server-name*
3. Specify a VPN instance for the virtual server.
   vpn-instance *vpn-instance-name*

   By default, a virtual server belongs to the public network.

Specifying link groups

About this task

When the primary link group is available (contains available links), the virtual server forwards packets through the primary link group. When the primary link group is not available, the virtual server forwards packets through the backup link group.

Procedure

1. Enter system view.
   system-view
2. Enter virtual server view.
   virtual-server *virtual-server-name*
3. Specify link groups.
   default link-group *link-group-name* [**backup** *backup-link-group-name*] [**sticky** *sticky-name*]

   By default, no link group is specified for a virtual server.

Specifying an LB policy

About this task

By referencing an LB policy, the virtual server load balances matching packets based on the packet contents.

Procedure

1. Enter system view.
   system-view
2. Enter virtual server view.
   virtual-server *virtual-server-name*
3. Specify an LB policy for the virtual server.
   lb-policy *policy-name*

   By default, the virtual server does not reference any LB policies.

   A virtual server can only reference a policy profile of the specified type. For example, a virtual server of the link-IP type can only reference a policy profile of the link-generic type.

Specifying a parameter profile

About this task

You can configure advanced parameters through a parameter profile. The virtual server references the parameter profile to analyze, process, and optimize service traffic.

Procedure

1. Enter system view.

   system-view

2. Enter virtual server view.

   virtual-server *virtual-server-name*

3. Specify a parameter profile for the virtual server.

   parameter ip *profile-name*

   By default, the virtual server does not reference any parameter profiles.

Configuring the bandwidth and connection parameters

1. Enter system view.

   system-view

2. Enter virtual server view.

   virtual-server *virtual-server-name*

3. Set the maximum bandwidth for the virtual server.

   rate-limit bandwidth [**inbound** | **outbound**] *bandwidth-value* **kbps**

   By default, the maximum bandwidth, inbound bandwidth, and outbound bandwidth for the virtual server are 0. The bandwidths are not limited.

4. Set the maximum number of connections for the virtual server.

   connection-limit max *max-number*

   By default, the maximum number of connections of the virtual server is 0. The number is not limited.

5. Set the maximum number of connections per second for the virtual server.

   rate-limit connection *connection-number*

   By default, the maximum number of connections per second for the virtual server is 0. The number is not limited.

Enabling the link protection feature

About this task

The link protection feature limits packets that exceed the bandwidth busy rate in the inbound or outbound direction of a link. When a link is busy in only the outbound direction, new traffic (traffic that does not match any sticky entry) is not distributed to the link. However, existing traffic is still distributed to the link. When a link is busy in only the inbound direction, new traffic can be distributed to the link.

When a link becomes busy in at least one direction, the link state becomes busy. When a link is not busy in both directions, the link sate is normal.

Restrictions and guidelines

This feature takes effect only when bandwidth statistics collection by interfaces is enabled.

Procedure

1. Enter system view.

   system-view

2. Enter virtual server view.

   virtual-server *virtual-server-name*

3. Enable the link protection feature.

   bandwidth busy-protection enable

   By default, the link protection feature is disabled.

Enabling bandwidth statistics collection by interfaces

About this task

By default, the load balancing module automatically collects link bandwidth statistics. Perform this task to enable interfaces to collect bandwidth statistics.

Procedure

1. Enter system view.

   system-view

2. Enter virtual server view.

   virtual-server *virtual-server-name*

3. Enable bandwidth statistics collection by interfaces.

   bandwidth interface statistics enable

   By default, bandwidth statistics collection by interfaces is disabled.

Specifying a DPI application profile

About this task

This task allows you to perform DPI on the traffic of a virtual server, including IPS, anti-virus,. DPI helps you identify network attacks and security risks to secure the LB device and internal servers. For more information about DPI application profiles, see DPI engine in *DPI Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter virtual server view.

   virtual-server *virtual-server-name*

3. Specify a DPI application profile for the virtual server.

   dpi-app-profile *dpi-app-profile-name*

   By default, a virtual server does not reference any DPI application profiles.

Configuring hot backup

About this task

To implement hot backup for two LB devices, you must enable synchronization for session extension information and sticky entries to avoid service interruption.

Restrictions and guidelines

For successful sticky entry synchronization, if you want to specify a sticky group, enable sticky entry synchronization before specifying a sticky group on both LB devices. You can specify a sticky group by using the **sticky** *sticky-name* option when specifying link groups.

In a VRRP network, you must specify the **global** keyword for the sticky entry synchronization feature to take effect.

The following configuration changes will cause the device to delete existing sticky entries and generate new ones based on subsequent traffic:

- Disable sticky entry synchronization.
- Change the sticky entry synchronization type.

Procedure

1. Enter system view.
   system-view
2. Enter virtual server view.
   virtual-server *virtual-server-name*
3. Enable session extension information synchronization.
   connection-sync enable
   By default, session extension information synchronization is disabled.
4. Enable sticky entry synchronization.
   sticky-sync enable [**global**]
   By default, sticky entry synchronization is disabled.

Enabling a virtual server

About this task

After you configure a virtual server, you must enable the virtual server for it to work.

Procedure

1. Enter system view.
   system-view
2. Enter virtual server view.
   virtual-server *virtual-server-name*
3. Enable the virtual server.
   service enable
   By default, the virtual server is disabled.

Configuring an LB class

An LB class classifies packets by comparing packets against specific rules. Matching packets are further processed by LB actions. You can create a maximum of 65535 rules for an LB class.

LB class tasks at a glance

To configure an LB class, perform the following tasks:

1. Creating an LB class

2. Creating a match rule

   Choose the following tasks as needed:

   - Creating a match rule that references an LB class
   - Creating a source IP address match rule
   - Creating a destination IP address match rule
   - Creating an ACL match rule
   - Creating an input interface match rule
   - Creating a user match rule
   - Creating a user group match rule
   - Creating a domain name match rule
   - Creating an ISP match rule
   - Creating an application group match rule

Creating an LB class

1. Enter system view.

   system-view

2. Create a link-generic LB class, and enter LB class view.

   loadbalance class *class-name* **type link-generic** [**match-all** | **match-any**]

   When you create an LB class, you must specify the class type. You can enter an existing LB class view without specifying the class type. If you specify the class type when entering an existing LB class view, the class type must be the one specified when you create the LB class.

3. (Optional.) Configure a description for the LB class.

   description *text*

   By default, no description is configured for the LB class.

Creating a match rule that references an LB class

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a match rule that references an LB class.

   match [*match-id*] **class** *class-name*

Creating a source IP address match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a source IP address match rule.

   match [*match-id*] **source** { **ip address** *ipv4-address* [*mask-length* | *mask*] | **ipv6 address** *ipv6-address* [*prefix-length*] }

Creating a destination IP address match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a destination IP address match rule.

   match [*match-id*] **destination** { **ip address** *ipv4-address* [*mask-length* | *mask*] | **ipv6 address** *ipv6-address* [*prefix-length*] }

Creating an ACL match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create an ACL match rule.

   match [*match-id*] **acl** [**ipv6**] { *acl-number* | **name** *acl-name* }

Creating an input interface match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create an input interface match rule.

   match [*match-id*] **interface** *interface-type interface-number*

Creating a user match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a user match rule.

   match [*match-id*] [**identity-domain** *domain-name*] **user** *user-name*

Creating a user group match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a user group match rule.

match [*match-id*] [**identity-domain** *domain-name*] **user-group** *user-group-name*

Creating a domain name match rule

About this task

The LB device stores mappings between domain names and IP addresses in the DNS cache. If the destination IP address of an incoming packet matches an IP address in the DNS cache, the LB device queries the domain name for the IP address. If the queried domain name matches the domain name configured in a match rule, the LB device takes the LB action on the packet.

Procedure

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a domain name match rule.

   match [*match-id*] **destination domain-name** *domain-name*

   By default, an LB class does not have any match rules.

Creating an ISP match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create an ISP match rule.

   match [*match-id*] **isp** *isp-name*

Creating an application group match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create an application group match rule.

   match [*match-id*] **app-group** *group-name*

   After you execute this configuration, hardware fast forwarding cannot process traffic processed by outbound link load balancing.

Configuring an LB action

About LB actions

LB actions include the following modes:

- **Forwarding mode**—Determines whether and how to forward packets. If no forwarding action is specified, packets are dropped.
- **Modification mode**—Modifies packets. To prevent the LB device from dropping the modified packets, the modification action must be used together with a forwarding action.

If you create an LB action without specifying any of the previous action modes, packets are dropped.

Restrictions and guidelines

The "Configuring the forwarding mode" and "Specifying link groups" tasks are mutually exclusive. Configuring one task automatically cancels the other task that you have configured.

LB action tasks at a glance

To configure an LB action, perform the following tasks:

1. Creating an LB action
2. (Optional.) Configuring a forwarding LB action
   - Configuring the forwarding mode
   - Specifying link groups
   - (Optional.) Matching the next rule upon failure to find a link
   - (Optional.) Matching the next rule when all links are busy
3. (Optional.) Configuring a modification LB action
   - Configuring the ToS field in IP packets sent to the server

Creating an LB action

1. Enter system view.

   system-view

2. Create a link-generic LB action and enter LB action view.

   loadbalance action *action-name* **type link-generic**

   When you create an LB action, you must specify the action type. You can enter an existing LB action view without specifying the action type. If you specify the action type when entering an existing LB action view, the action type must be the one specified when you create the LB action.

3. (Optional.) Configure a description for the LB action.

   description *text*

   By default, no description is configured for the LB action.

Configuring a forwarding LB action

About this task

Three forwarding LB action types are available:

- **Forward**—Forwards matching packets.
- **Specify link groups**—When the primary link group is available (contains available real servers), the primary link group is used to guide packet forwarding. When the primary link group is not available, the backup link group is used to guide packet forwarding.
- **Match the next rule upon failure to find a link**—If the device fails to find a link according to the LB action, it matches the packet with the next rule in the LB policy.

- Match the next rule when all links are busy.

Configuring the forwarding mode

1. Enter system view.

   system-view

2. Enter LB action view.

   loadbalance action *action-name*

3. Configure the forwarding mode.

   forward all

   By default, the forwarding mode is to discard packets.

Specifying link groups

1. Enter system view.

   system-view

2. Enter LB action view.

   loadbalance action *action-name*

3. Specify link groups.

   link-group *link-group-name* [**backup** *backup-link-group-name*] [**sticky** *sticky-name*]

   By default, no link group is specified.

Matching the next rule upon failure to find a link

1. Enter system view.

   system-view

2. Enter LB action view.

   loadbalance action *action-name*

3. Match the next rule upon failure to find a link.

   fallback-action continue

   By default, the next rule is not matched when no links are available for the current LB action.

   This command does not apply to SIP virtual servers.

Matching the next rule when all links are busy

1. Enter system view.

   system-view

2. Enter LB action view.

   loadbalance action *action-name*

3. Match the next rule when all links are busy.

   busy-action continue

   By default, the device assigns packets to links regardless of whether they are busy.

Configuring the ToS field in IP packets sent to the server

1. Enter system view.

   system-view

2. Enter LB action view.

   loadbalance action *action-name*

3. Configure the ToS field in IP packets sent to the server.

set ip tos *tos-number*

By default, the ToS field in IP packets sent to the server is not changed.

Configuring an LB policy

About LB policies

An LB policy associates an LB class with an LB action to guide packet forwarding. In an LB policy, you can configure an LB action for packets matching the specified LB class, and configure the default action for packets matching no LB class.

You can specify multiple LB classes for an LB policy. Packets match the LB classes in the order the LB classes are configured. If an LB class is matched, the specified LB action is performed. If no LB class is matched, the default LB action is performed.

LB policy tasks at a glance

To configure an LB policy, perform the following tasks:

1. Creating an LB policy
2. Specifying an LB action
3. Specifying the default LB action

Creating an LB policy

1. Enter system view.

   system-view

2. Create a link-generic LB policy, and enter LB action view.

   loadbalance policy *policy-name* **type link-generic**

   When you create an LB policy, you must specify the policy type. You can enter an existing LB policy view without specifying the policy type. If you specify the policy type when entering an existing LB policy view, the policy type must be the one specified when you create the LB policy.

3. (Optional.) Configure a description for the LB policy.

   description *text*

   By default, no description is configured for an LB policy.

Specifying an LB action

Restrictions and guidelines

A link-generic LB policy can reference only link-generic LB classes and link-generic LB actions.

Procedure

1. Enter system view.

   system-view

2. Enter LB policy view.

   loadbalance policy *policy-name*

3. Specify an LB action for the specified LB class.

   class *class-name* [**insert-before** *before-class-name* | **insert-after** [*after-class-name*]] **action** *action-name*

By default, no LB action is specified for any LB classes.

You can specify an LB action for different LB classes.

Specifying the default LB action

Restrictions and guidelines

A link-generic LB policy can only reference link-generic LB actions.

Procedure

1. Enter system view.

   `system-view`

2. Enter LB policy view.

   `loadbalance policy` *`policy-name`*

3. Specify the default LB action.

   `default-class action` *`action-name`*

   By default, no default LB action is specified.

Configuring a sticky group

A sticky group uses a sticky method to distribute similar sessions to the same link according to sticky entries. The sticky method applies to the first packet of a session. Other packets of the session are distributed to the same link.

Sticky group tasks at a glance

To configure a sticky group, perform the following tasks:

1. Creating a sticky group
2. Configuring the IP sticky method
3. (Optional.) Configuring the timeout time for sticky entries
4. (Optional.) Ignoring the limits for sessions that match sticky entries
5. (Optional.) Enabling stickiness-over-busyness

Creating a sticky group

1. Enter system view.

   `system-view`

2. Create an address- and port-type sticky group and enter sticky group view.

   `sticky-group` *`group-name`* **`type address-port`**

   When you create a sticky group, you must specify the group type. You can enter an existing sticky group view without specifying the group type. If you specify the group type when entering an existing sticky group view, the group type must be the one specified when you create the sticky group.

3. (Optional.) Configure a description for the sticky group.

   `description` *`text`*

   By default, no description is configured for the sticky group.

Configuring the IP sticky method

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Configure the IP sticky method.

   IPv4:

   ip [**port**] { **both** | **destination** | **source** } [**mask** *mask-length*]

   IPv6:

   ipv6 [**port**] { **both** | **destination** | **source** } [**prefix** *prefix-length*]

   By default, no IP sticky method is configured.

Configuring the timeout time for sticky entries

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Configure the timeout time for sticky entries.

   timeout *timeout-value*

   By default, the timeout time for sticky entries is 60 seconds.

Ignoring the limits for sessions that match sticky entries

About this task

Perform this task to ignore the following limits for sessions that match sticky entries:

- Bandwidth and connection parameters on links.
- LB connection limit policies on virtual servers.

Procedure

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Ignore the limits for sessions that match sticky entries.

   override-limit enable

   By default, the session limits apply to sessions that match sticky entries.

Enabling stickiness-over-busyness

About stickiness-over-busyness

This feature enables the device to assign client requests to links based on sticky entries, regardless of whether the links are busy.

When this feature is disabled, the device assigns client requests to only links in normal state.

Procedure

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Enable stickiness-over-busyness.

   sticky-over-busy enable

   By default, stickiness-over-busyness is disabled.

Configuring a parameter profile

About configuring a parameter profile

You can configure advanced parameters through a parameter profile. The virtual server references the parameter profile to analyze, process, and optimize service traffic.

Creating a parameter profile

1. Enter system view.

   system-view

2. Create an IP-type parameter profile and enter parameter profile view.

   parameter-profile *profile-name* **type ip**

   By default, no parameter profiles exist.

   When you create a parameter profile, you must specify the profile type. You can enter an existing parameter profile view without specifying the profile type. If you specify the profile type when entering an existing parameter profile view, the profile type must be the one specified when you create the parameter profile.

3. (Optional.) Configure a description for the parameter profile.

   description *text*

   By default, no description is configured for the parameter profile.

Configuring the ToS field in IP packets sent to the client

1. Enter system view.

   system-view

2. Enter IP parameter profile view.

   parameter-profile *profile-name*

3. Configure the ToS field in IP packets sent to the client.

   set ip tos *tos-number*

   By default, the ToS field in IP packets sent to the client is not changed.

Configuring ISP information

About configuring ISP information

Perform this task to configure IP address information for an ISP. The IP address information can be used by an ISP match rule. When the destination IP address of packets matches the ISP match rule of an LB class, the LB device takes the action associated with the class. The device supports the following methods to configure IP address information:

- **Manual configuration**—The administrator manually specifies IP address information.
- **ISP auto update**—With ISP auto update enabled, the device regularly queries IP address information from the whois server according to the whois maintainer object of the ISP.
- **ISP file import**—The administrator manually imports an ISP file in .tp format. The ISP file can be obtained from the official website.

Restrictions and guidelines

You can configure ISP information manually, by importing an ISP file, by auto update, or use the combination of these methods..

Configuring ISP information manually

1. Enter system view.

   `system-view`

2. Create an ISP and enter ISP view.

   `loadbalance isp name` *`isp-name`*

3. Specify the IP address for the ISP.

   IPv4:

   `ip address` *`ipv4-address`* { *`mask-length`* | *`mask`* }

   IPv6:

   `ipv6 address` *`ipv6-address prefix-length`*

   By default, an ISP does not contain IPv4 or IPv6 addresses.

   An ISP does not allow overlapping network segments.

4. (Optional.) Configure a description for the ISP.

   `description` *`text`*

   By default, no description is configured for the ISP.

Configuring ISP auto update

1. Enter system view.

   `system-view`

2. Create an ISP and enter ISP view.

   `loadbalance isp name` *`isp-name`*

3. Specify a whois maintainer object for the ISP.

   `whois-mntner` *`mntner-name`*

   By default, no whois maintainer object is specified.

   You can specify a maximum of 10 whois maintainer objects for an ISP.

4. Return to system view.

   quit

5. Enable ISP auto update.

   loadbalance isp auto-update enable

   By default, ISP auto update is disabled.

6. Configure the ISP auto update frequency.

   loadbalance isp auto-update frequency { *per-day* | *per-week* | *per-month* }

   By default, the ISP auto update is performed once per week.

7. Specify the whois server to be queried for ISP auto update.

   loadbalance isp auto-update whois-server { **domain** *domain-name* | **ip** *ip-address* }

   By default, no whois server is specified for ISP auto update.

Importing an ISP file

1. Enter system view.

   system-view

2. Import an ISP file.

   loadbalance isp file *isp-file-name*

Setting the aging time for DNS cache entries

About this task

A DNS cache entry records the mapping between a domain name and the IP address of the outbound next hop.

Procedure

1. Enter system view.

   system-view

2. Set the aging time for DNS cache entries.

   loadbalance dns-cache aging-time *aging-time*

   By default, the aging time for DNS cache entries is 60 minutes.

Configuring the ALG feature

About this task

The Application Level Gateway (ALG) feature distributes parent and child sessions to the same link.

Procedure

1. Enter system view.

   system-view

2. Enable ALG.

   - Enable ALG for the specified protocol:
     loadbalance alg { **dns** | **ftp** | **h323** | **icmp-error** | **ils** | **mgcp** | **nbt** | **pptp** | **rsh** | **rtsp** | **sccp** | **sip** | **sqlnet** | **tftp** | **xdmcp** }

- Enable ALG for all protocols:
  `loadbalance alg all-enable`

By default, ALG is enabled for the DNS, FTP, PPTP, and RTSP protocols and ICMP error packets.

Performing a load balancing test

About performing a load balancing test

Perform this task in any view to test the load balancing result.

Performing an IPv4 load balancing test

To perform an IPv4 load balancing test, execute the following command in any view:

loadbalance schedule-test ip [**vpn-instance** *vpn-instance-name*] { **application http** { **message-file** *file-name* | **method** { **get** | **post** } **url** *url* [**header** *header*]&<1-10> [**content** *content-value*] } | **protocol** { *protocol-number* | **icmp** | **tcp** | **udp** } } **destination** *destination-address* **destination-port** *destination-port* **source** *source-address* **source-port** *source-port* [**slot** *slot-number*]

Performing an IPv6 load balancing test

To perform an IPv6 load balancing test, execute the following command in any view:

loadbalance schedule-test ipv6 [**vpn-instance** *vpn-instance-name*] { **application http** { **message-file** *file-name* | **method** { **get** | **post** } **url** *url* [**header** *header*]&<1-10> [**content** *content-value*] } | **protocol** { *protocol-number* | **icmpv6** | **tcp** | **udp** } } **destination** *destination-address* **destination-port** *destination-port* **source** *source-address* **source-port** *source-port* [**slot** *slot-number*]

Enabling SNMP notifications

About this task

To report critical load balancing events to an NMS, enable SNMP notifications for load balancing. For load balancing event notifications to be sent correctly, you must also configure SNMP as described in *Network Management and Monitoring Configuration Guide*.

The SNMP notifications configuration tasks for Layer 4 and Layer 7 server load balancing are the same.

Procedure

1. Enter system view.
   `system-view`
2. Enable SNMP notifications for load balancing.
   `snmp-agent trap enable loadbalance`
   By default, SNMP notifications are enabled for load balancing.

Enabling load balancing logging

About load balancing logging

For security auditing purposes, enable load balancing logging to record load balancing information. Load balancing logging includes the following types:

- Basic logging.
- Link flow logging.
- NAT logging.
- Link busy state logging.

Basic logging generates logs for the following events:

- The state of a link or link group changes.
- The health monitoring result of a link changes.
- The number of connections on a link or virtual server reaches or drops below the upper limit.
- The connection establishment rate on a link or virtual server reaches or drops below the upper limit.
- A primary/backup server farm switchover occurs between server farms specified for a virtual server.
- A primary/backup server farm switchover occurs between server farms specified for an LB action.

Link flow logging records flows forwarded through all links.

NAT logging records NAT session information, including IP address and port translation information and access information.

Link busy state logging records busy states for all links.

Enabling load balancing basic logging

1. Enter system view.
   ```
   system-view
   ```
2. Enable load balancing basic logging.
   ```
   loadbalance log enable base
   ```
   By default, load balancing basic logging is enabled.

Enabling load balancing link flow logging

1. Enter system view.
   ```
   system-view
   ```
2. Enable load balancing link flow logging.
   ```
   loadbalance log enable link-flow
   ```
   By default, load balancing link flow logging is enabled.

Enabling load balancing NAT logging

1. Enter system view.
   ```
   system-view
   ```

2. Enable load balancing NAT logging.

   loadbalance log enable nat

   By default, load balancing NAT logging is disabled.

Enabling load balancing link busy state logging

1. Enter system view.

   system-view

2. Enable load balancing link busy state logging.

   loadbalance log enable bandwidth-busy

   By default, load balancing link busy state logging is disabled.

Displaying and maintaining outbound link load balancing

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display LB action information.	**display loadbalance action** [**name** *action-name*]
Display LB class information.	**display loadbalance class** [**name** *class-name*]
Display LB hot backup statistics.	**display loadbalance hot-backup statistics** [**slot** *slot-number*]
Display ISP information.	**display loadbalance isp** [**ip** *ipv4-address* | **ipv6** *ipv6-address* | **name** *isp-name*]
Display LB policy information.	**display loadbalance policy** [**name** *policy-name*]
Display proximity entry information.	**display loadbalance proximity** [**vpn-instance** *vpn-instance-name*] [**ip** [*ipv4-address*] | **ipv6** [*ipv6-address*]] [**slot** *slot-number*]
Display parameter profile information.	**display parameter-profile** [**name** *parameter-name*]
Display link information.	**display loadbalance link** [**brief** | **name** *link-name*]
Display link group member information.	**display loadbalance link link-group** *link-group-name* [**name** *link-name*]
Display link statistics.	**display loadbalance link statistics** [**name** *link-name*] [**slot** *slot-number*]
Display link group member statistics.	**display loadbalance link statistics link-group** *link-group-name* [**name** *link-name*] [**slot** *slot-number*]
Display link outbound interface statistics.	**display loadbalance link out-interface statistics** [**name** *link-name*]
Display link group information.	**display loadbalance link-group** [**brief** | **name**

Task	Command
	link-group-name]
Display sticky entry information.	**display sticky virtual-server** [*virtual-server-name*] [**class** *class-name* | **default-class** | **default-link-group**] [**slot** *slot-number*]
Display sticky group information.	**display sticky-group** [**name** *group-name*]
Display virtual server information.	**display virtual-server** [**brief** | **name** *virtual-server-name*]
Display virtual server statistics.	**display virtual-server statistics** [**name** *virtual-server-name*] [**slot** *slot-number*]
Display the ALG status for all protocols.	**display loadbalance alg**
Display DNS cache information.	**display loadbalance dns-cache** [**vpn-instance** *vpn-instance-name*] [**domain-name** *domain-name*] [**slot** *slot-number*]
Clear LB hot backup statistics.	**reset loadbalance hot-backup statistics**
Clear proximity entry information.	**reset loadbalance proximity** [**vpn-instance** *vpn-instance-name*] [**ip** [*ipv4-address*] | **ipv6** [*ipv6-address*]]
Clear all Layer 7 connections.	**reset loadbalance connections**
Clear link statistics.	**reset loadbalance link statistics** [*link-name*]
Clear link group member statistics.	**reset loadbalance link statistics link-group** *link-group-name* [**name** *link-name*]
Clear virtual server statistics.	**reset virtual-server statistics** [*virtual-server-name*]
Clear DNS cache information.	**reset loadbalance dns-cache** [**vpn-instance** *vpn-instance-name*] [**domain-name** *domain-name*]

Outbound link load balancing configuration examples

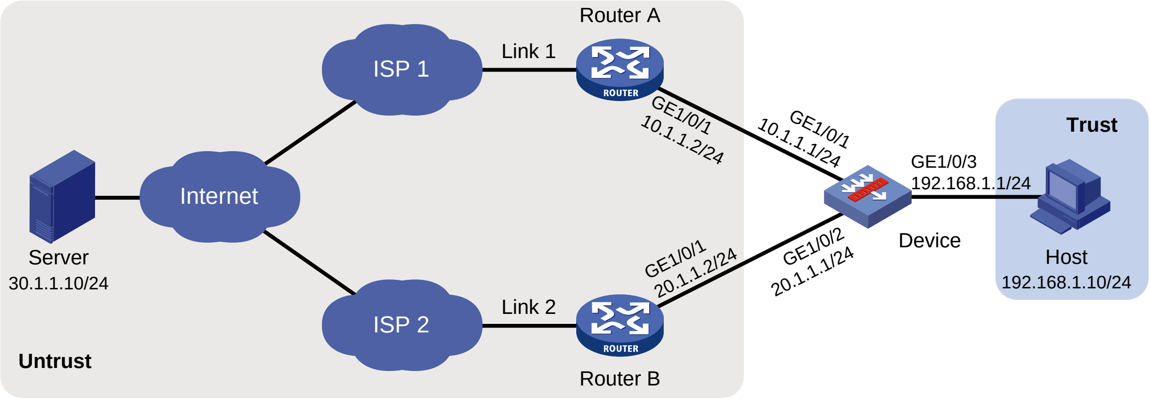
Example: Configuring outbound link load balancing

Network configuration

In Figure 4, ISP 1 and ISP 2 provide two links, Link 1 and Link 2, with the same router hop count, bandwidth, and cost. Link 1 has lower network delay.

Configure link load balancing for the device to select an optimal link for traffic from the client host to the server.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/3
   [Device-security-zone-Trust] quit
   ```

3. Configure a security policy:

   Configure rules to permit traffic from the **Trust** security zone to the **Untrust** security zone and traffic from the **Local** security zone to the **Untrust** security zone, so the users can access the server:

   # Configure a rule named **lbrule1** to allow the users to access the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name lbrule1
   [Device-security-policy-ip-1-lbrule1] source-zone trust
   [Device-security-policy-ip-1-lbrule1] destination-zone untrust
   [Device-security-policy-ip-1-lbrule1] source-ip-subnet 192.168.1.0 255.255.255.0
   [Device-security-policy-ip-1-lbrule1] action pass
   [Device-security-policy-ip-1-lbrule1] quit
   ```

   # Configure a rule named **lblocalout** to allow the device to send probe packets to the next hop.

   ```
   [Device-security-policy-ip] rule name lblocalout
   [Device-security-policy-ip-2-lblocalout] source-zone local
   [Device-security-policy-ip-2-lblocalout] destination-zone untrust
   [Device-security-policy-ip-2-lblocalout] destination-ip-subnet 10.1.1.0
   255.255.255.0
   ```

```
[Device-security-policy-ip-2-lblocalout] destination-ip-subnet 20.1.1.0
255.255.255.0
[Device-security-policy-ip-2-lblocalout] action pass
[Device-security-policy-ip-2-lblocalout] quit
[Device-security-policy-ip] quit
```

4. Configure a link group:

   # Create the ICMP-type NQA template **t1**, and configure the NQA client to send the probe result to the feature that uses the template on a per-probe basis.

   ```
   [Device] nqa template icmp t1
   [Device-nqatplt-icmp-t1] reaction trigger per-probe
   [Device-nqatplt-icmp-t1] quit
   ```

   # Specify the default proximity probe method as **t1**, and set the network delay weight for proximity calculation to 200.

   ```
   [Device] loadbalance proximity
   [Device-lb-proximity] match default probe t1
   [Device-lb-proximity] rtt weight 200
   [Device-lb-proximity] quit
   ```

   # Create the link group **lg**, and enable the proximity feature.

   ```
   [Device] loadbalance link-group lg
   [Device-lb-lgroup-lg] proximity enable
   ```

   # Disable the NAT feature.

   ```
   [Device-lb-lgroup-lg] transparent enable
   [Device-lb-lgroup-lg] quit
   ```

5. Configure links:

   # Create the link **link1** with next hop address 10.1.1.2, and add it to the link group **lg**.

   ```
   [Device] loadbalance link link1
   [Device-lb-link-link1] router ip 10.1.1.2
   [Device-lb-link-link1] link-group lg
   [Device-lb-link-link1] quit
   ```

   # Create the link **link2** with next hop address 20.1.1.2, and add it to link group **lg**.

   ```
   [Device] loadbalance link link2
   [Device-lb-link-link2] router ip 20.1.1.2
   [Device-lb-link-link2] link-group lg
   [Device-lb-link-link2] quit
   ```

6. Create the link-IP virtual server **vs** with VSIP 0.0.0.0/0, specify its default master link group **lg**, and enable the virtual server.

   ```
   [Device] virtual-server vs type link-ip
   [Device-vs-link-ip-vs] virtual ip address 0.0.0.0 0
   [Device-vs-link-ip-vs] default link-group lg
   [Device-vs-link-ip-vs] service enable
   [Device-vs-link-ip-vs] quit
   ```

Verifying the configuration

# Display brief information about all links.

```
[Device] display loadbalance link brief
Link             Route IP            State        VPN instance   Link group
link1            10.1.1.2            Active                      lg
link2            20.1.1.2            Active                      lg
```

# Display detailed information about all link groups.

```
[Device] display loadbalance link-group
Link group: lg
  Description:
  Predictor: Round robin
  Proximity: Enabled
  NAT: Disabled
  SNAT pool:
  Failed action: Keep
  Active threshold: Disabled
  Slow-online: Disabled
  Selected link: Disabled
  Probe information:
    Probe success criteria: All
    Probe method:
    t1
  Total link: 2
  Active link: 2
  Link list:
  Name          State         VPN instance  Router IP             Weight Priority
  link1         Active                      10.1.1.2              100    4
  link2         Active                      20.1.1.2              100    4
```

# Display detailed information about all virtual servers.

```
[Device] display virtual-server
Virtual server: vs
  Description:
  Type: LINK-IP
  State: Active
  VPN instance:
  Virtual IPv4 address: 0.0.0.0/0
  Virtual IPv6 address: --
  Port: 0
  Primary link group: lg (in use)
  Backup link group:
  Sticky:
  LB policy:
  LB limit-policy:
  Connection limit: --
  Rate limit:
    Connections: --
    Bandwidth: --
    Inbound bandwidth: --
    Outbound bandwidth: --
  Connection synchronization: Disabled
  Sticky synchronization: Disabled
  Bandwidth busy protection: Disabled
  Interface bandwidth statistics: Disabled
  Route advertisement: Disabled
```

# Display brief information about all IPv4 proximity entries.

```
[Device] display loadbalance proximity ip
  IPv4 entries in total: 1
    IPv4 address/Mask length       Timeout     Best link
    ------------------------------------------------------------
    10.1.0.0/24                    50          link1
```

Configuring transparent DNS proxies

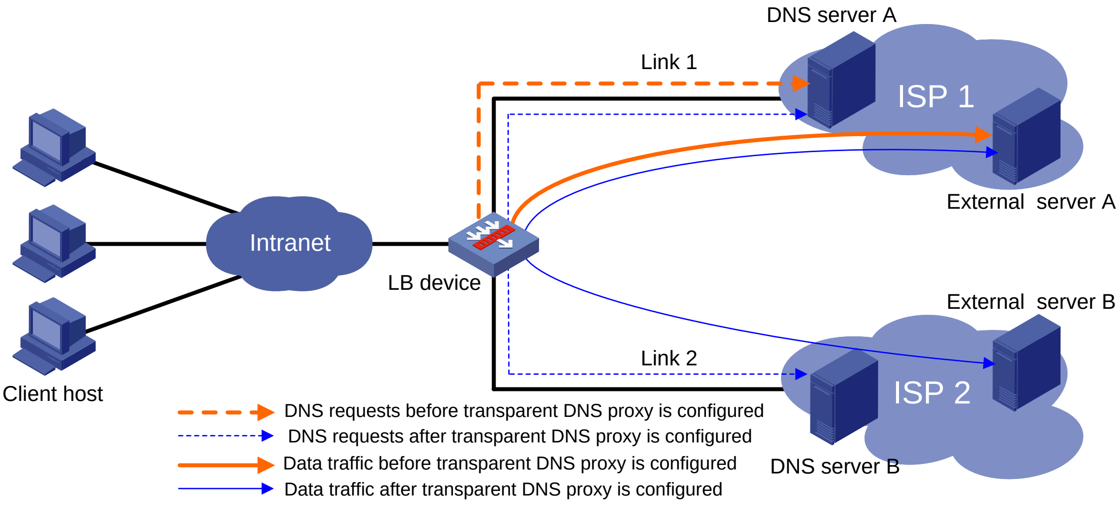
About transparent DNS proxies

Application scenario

As shown in Figure 5, intranet users of an enterprise can access external servers A and B through link 1 of ISP 1 and link 2 of ISP 2. External servers A and B provide the same services. All DNS requests of intranet users are forwarded to DNS server A, which returns the resolved IP address of external server A to the requesting users. In this way, all traffic of intranet users is forwarded on one link. Link congestion might occur.

The transparent DNS proxy feature can solve this problem by forwarding DNS requests to DNS servers in different ISPs. All traffic from intranet users is evenly distributed on multiple links. This feature can prevent link congestion and ensure service continuity upon a link failure.

Figure 5 Transparent DNS proxy working mechanism


DNS server A
Link 1
ISP 1
External server A
Intranet
LB device
External server B
Link 2
ISP 2
Client host
DNS server B
DNS requests before transparent DNS proxy is configured
DNS requests after transparent DNS proxy is configured
Data traffic before transparent DNS proxy is configured
Data traffic after transparent DNS proxy is configured

Workflow

The transparent DNS proxy is implemented by changing the destination IP address of DNS requests.

Figure 6 Transparent DNS proxy workflow

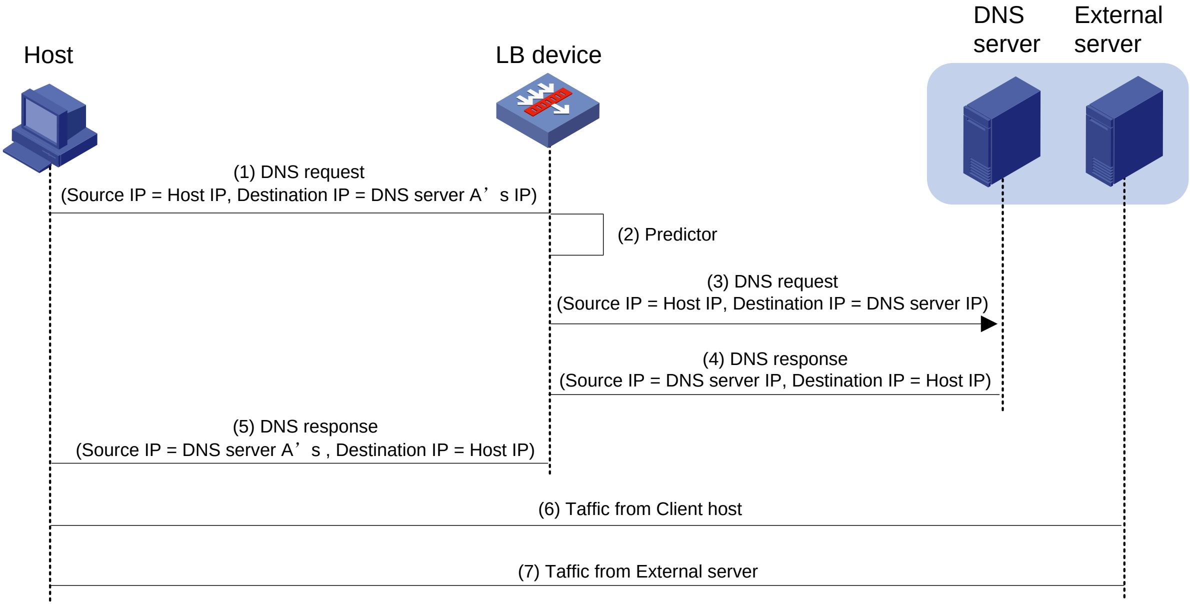

Host
LB device
DNS server
External server
(1) DNS request
(Source IP = Host IP, Destination IP = DNS server A’ s IP)
(2) Predictor
(3) DNS request
(Source IP = Host IP, Destination IP = DNS server IP)
(4) DNS response
(Source IP = DNS server IP, Destination IP = Host IP)
(5) DNS response
(Source IP = DNS server A’ s , Destination IP = Host IP)
(6) Taffic from Client host
(7) Taffic from External server

Table 1 Workflow description

Step	Source IP address	Destination IP address
1. An intranet user on the client host sends a DNS request to the LB device.	Host IP address	IP address of DNS server A
2. The LB device selects a DNS server to forward the DNS request according to the scheduling algorithm.	N/A	N/A
3. The LB device changes the destination IP address of the DNS request as the IP address of the selected DNS server.	Host IP address	IP address of the selected DNS server
4. The DNS server processes the DNS request and replies with a DNS response.	IP address of the selected DNS server	Host IP address
5. The LB device changes the source IP address of the DNS response as the destination IP address of the DNS request.	IP address of DNS server A	Host IP address
6. The intranet user accesses the external server according to the resolved IP address in the DNS response.	Host IP address	IP address of the external server
7. The external server responds to the intranet user.	IP address of the external server	Host IP address

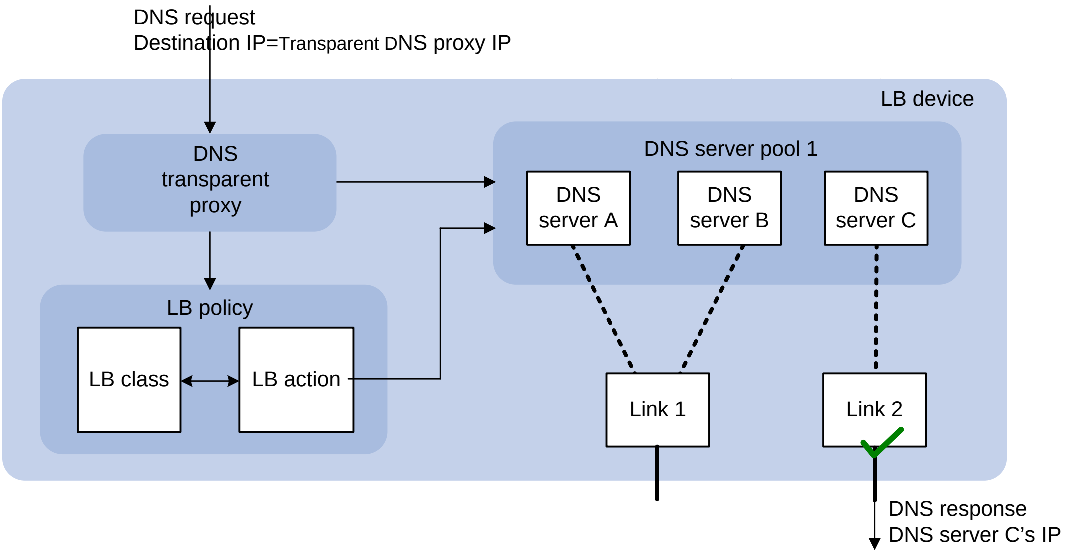
Transparent DNS proxy on the LB device

The LB device distributes DNS requests to multiple links by changing the destination IP address of DNS requests.

As shown in Figure 7, the LB device contains the following elements:

- **Transparent DNS proxy**—The LB device performs transparent DNS proxy for a DNS request only when the port number of the DNS request matches the port number of the transparent DNS proxy.
- **DNS server pool**—A group of DNS servers.
- **DNS server**—Entity that processes DNS requests.
- **Link**—Physical link provided by an ISP.
- **LB class**—Classifies packets to implement load balancing based on packet type.
- **LB action**—Drops, forwards, or modifies packets.
- **LB policy**—Associates an LB class with an LB action. An LB policy can be referenced by the transparent DNS proxy.

Figure 7 Transparent DNS proxy on the LB device



If the destination IP address and port number of a DNS request match those of the transparent DNS proxy, the LB device processes the DNS request as follows:

1. The LB device finds the DNS server pool associated with the transparent DNS proxy.
2. The LB device selects a DNS server according to the scheduling algorithm configured for the DNS server pool.
3. The LB device uses the IP address of the selected DNS server as the destination IP address of the DNS request, and sends the request to the DNS server.
4. The DNS server receives and processes the DNS request, and replies with a DNS response.

   The intranet user can now access the external server after receiving the DNS response.

Transparent DNS proxy tasks at a glance

To configure the transparent DNS proxy feature, perform the following tasks:

1. Configuring a transparent DNS proxy
2. Configuring a DNS server pool
3. Configuring a DNS server
4. Configuring a link
5. (Optional.) Configuring an LB policy
   a. Configuring an LB class

    b. Configuring an LB action
    c. Configuring an LB policy
6. (Optional.) Configuring a sticky group
7. (Optional.) Enabling load balancing logging

Configuring a transparent DNS proxy

By configuring a transparent DNS proxy, you can load balance DNS requests that match the transparent DNS proxy.

Restrictions and guidelines

If both the "Specifying the default DNS server pool" and "Specifying an LB policy" tasks are configured, packets are processed by the LB policy first. If the processing fails, the packets are processed by the default DNS server pool.

Transparent DNS proxy tasks at a glance

To configure a transparent DNS proxy, perform the following tasks:

1. Creating a transparent DNS proxy
2. Specifying an IP address and port number
3. Configuring a packet processing policy
   Choose the following tasks as needed:
   - Specifying the default DNS server pool
   - Specifying an LB policy
4. (Optional.) Specifying a VPN instance
5. (Optional.) Enabling the link protection feature
6. (Optional.) Configuring hot backup
7. Enabling the transparent DNS proxy

Creating a transparent DNS proxy

1. Enter system view.
   `system-view`
2. Create a transparent DNS proxy and enter its view.
   `loadbalance dns-proxy` *`dns-proxy-name`* **`type udp`**

Specifying an IP address and port number

Restrictions and guidelines

As a best practice, configure an all-zero IP address for a transparent DNS proxy. In this case, all DNS requests are processed by the transparent DNS proxy.

Procedure

1. Enter system view.
   `system-view`
2. Enter transparent DNS proxy view.

```
loadbalance dns-proxy dns-proxy-name
```

3. Specify an IP address for the transparent DNS proxy.

   IPv4:

   ```
   ip address ipv4-address [ mask-length | mask ]
   ```

   IPv6:

   ```
   ipv6 address ipv6-address [ prefix-length ]
   ```

   By default, no IP address is specified for a transparent DNS proxy.

4. Specify the port number for the transparent DNS proxy.

   ```
   port port-number
   ```

   By default, the port number is 53 for a transparent DNS proxy.

Specifying the default DNS server pool

1. Enter system view.

   ```
   system-view
   ```

2. Enter transparent DNS proxy view.

   ```
   loadbalance dns-proxy dns-proxy-name
   ```

3. Specify the default DNS server pool for the transparent DNS proxy.

   ```
   default dns-server-pool pool-name [ sticky sticky-name ]
   ```

   By default, no default DNS server pool is specified for a transparent DNS proxy.

Specifying an LB policy

About this task

By referencing an LB policy, the transparent DNS proxy load balances matching DNS requests based on the packet contents. For more information about configuring an LB policy, see "Configuring an LB policy."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter transparent DNS proxy view.

   ```
   loadbalance dns-proxy dns-proxy-name
   ```

3. Specify an LB policy for the transparent DNS proxy.

   ```
   lb-policy policy-name
   ```

   By default, a transparent DNS proxy does not reference any LB policies.

Specifying a VPN instance

1. Enter system view.

   ```
   system-view
   ```

2. Enter transparent DNS proxy view.

   ```
   loadbalance dns-proxy dns-proxy-name
   ```

3. Specify a VPN instance for the transparent DNS proxy.

   ```
   vpn-instance vpn-instance-name
   ```

   By default, a transparent DNS proxy belongs to the public network.

Enabling the link protection feature

About this task

This feature enables a transparent DNS proxy to select a DNS server based on the link bandwidth ratio. If the bandwidth ratio of a link is exceeded, the DNS server is not selected.

If the traffic volume on the link to a DNS server exceeds the maximum expected bandwidth multiplied by the bandwidth ratio, the DNS server is busy and will not be selected. If the traffic volume drops below the maximum expected bandwidth multiplied by the bandwidth recovery ratio, the DNS server participates in scheduling again. For more information about setting the bandwidth ratio, see "Setting the bandwidth ratio and maximum expected bandwidth."

Procedure

1. Enter system view.

   `system-view`

2. Enter transparent DNS proxy view.

   `loadbalance dns-proxy` *`dns-proxy-name`*

3. Enable the link protection feature.

   `bandwidth busy-protection enable`

   By default, the link protection feature is disabled.

Configuring hot backup

About this task

To implement hot backup for two LB devices, you must enable synchronization for session extension information and sticky entries to avoid service interruption.

Restrictions and guidelines

For successful sticky entry synchronization, if you want to specify a sticky group, enable sticky entry synchronization before specifying a sticky group on both LB devices. You can specify a sticky group by using the **`sticky`** *`sticky-name`* option when specifying the default DNS server pool.

The following configuration changes will cause the device to delete existing sticky entries and generate new ones based on subsequent traffic:

- Disable sticky entry synchronization.
- Change the sticky entry synchronization type.

Procedure

1. Enter system view.

   `system-view`

2. Enter transparent DNS proxy view.

   `loadbalance dns-proxy` *`dns-proxy-name`*

3. Enable session extension information synchronization.

   `connection-sync enable`

   By default, session extension information synchronization is disabled.

4. Enable sticky entry synchronization.

   `sticky-sync enable`

   By default, sticky entry synchronization is disabled.

Enabling the transparent DNS proxy

About this task

After configuring a transparent DNS proxy, you must enable the transparent DNS proxy for it to work.

Procedure

1. Enter system view.
   system-view
2. Enter transparent DNS proxy view.
   loadbalance dns-proxy *dns-proxy-name*
3. Enable the transparent DNS proxy.
   service enable
   By default, a transparent DNS proxy is disabled.

Configuring a DNS server pool

By configuring a DNS server pool, you can perform centralized management on DNS servers that have similar functions.

Creating a DNS server pool

1. Enter system view.
   system-view
2. Create a DNS server pool and enter its view.
   loadbalance dns-server-pool *pool-name*
3. (Optional.) Configure a description for the DNS server pool.
   description *text*
   By default, no description is configured for a DNS server pool.

Adding and configuring a DNS server pool member

About this task

Perform this task to create a DNS server pool member or add an existing DNS server as a DNS server pool member in DNS server pool view. You can also specify a DNS server pool for a DNS server in DNS server view to achieve the same purpose (see "Creating a DNS server and specifying a DNS server pool").

After adding a DNS server pool member, you can configure the following parameters and features for the DNS server in the DNS server pool:

- Weight.
- Priority.
- Health monitoring.

The member-based scheduling algorithm selects the best DNS server based on these configurations.

Adding a DNS server pool member

1. Enter system view.

system-view

2. Enter DNS server pool view.

   loadbalance dns-server-pool *pool-name*

3. Create and add a DNS server pool member and enter DNS server pool member view.

   dns-server *dns-server-name* **port** *port-number*

   If the DNS server already exists, the command adds the existing DNS server as a DNS server pool member.

4. (Optional.) Configure a description for the DNS server pool member.

   description *text*

   By default, no description is configured for the DNS server pool member.

Setting the weight and priority of the DNS server pool member

1. Enter system view.

   system-view

2. Enter DNS server pool view.

   loadbalance dns-server-pool *pool-name*

3. Enter DNS server pool member view.

   dns-server *dns-server-name* **port** *port-number*

4. Set the weight of the DNS server pool member.

   weight *weight-value*

   The default setting is 100.

5. Set the priority of the DNS server pool member.

   priority *priority*

   The default setting is 4.

Configuring health monitoring for the DNS server pool member

1. Enter system view.

   system-view

2. Enter DNS server pool view.

   loadbalance dns-server-pool *pool-name*

3. Enter DNS server pool member view.

   dns-server *dns-server-name* **port** *port-number*

4. Specify a health monitoring method for the DNS server pool member.

   probe *template-name*

   By default, no health monitoring method is specified for the DNS server pool member.

   You can specify an NQA template for health monitoring. For information about NQA templates, see NQA configuration in *Network Management and Monitoring Configuration Guide*.

5. Specify the health monitoring success criteria for the DNS server pool member.

   success-criteria { **all** | **at-least** *min-number* }

   By default, health monitoring succeeds only when all the specified health monitoring methods succeed.

Configuring a scheduling algorithm for a DNS server pool

About this task

Perform this task to specify a scheduling algorithm for a DNS server pool and specify the number of DNS servers to participate in scheduling. The LB device calculates the DNS servers to process DNS requests based on the following scheduling algorithms:

- **Source IP address hash algorithm**—Hashes the source IP address of DNS requests and distributes DNS requests to different DNS servers according to the hash values. This hash algorithm ensures that DNS requests with the same source IP address are distributed to the same DNS server.
- **Source IP address and port hash algorithm**—Hashes the source IP address and port number of DNS requests and distributes DNS requests to different DNS servers according to the hash values. This hash algorithm ensures that DNS requests with the same source IP address and port number are distributed to the same DNS server.
- **Destination IP address hash algorithm**—Hashes the destination IP address of DNS requests and distributes DNS requests to different DNS servers according to the hash values. This hash algorithm ensures that DNS requests with the same destination IP address are distributed to the same DNS server.
- **Random algorithm**—Distributes DNS requests to DNS servers randomly.
- **Weighted round-robin algorithm**—Distributes DNS requests to DNS servers in a round-robin manner according to the weights of DNS servers. For example, you can assign weight values 2 and 1 to DNS server A and DNS server B, respectively. This algorithm distributes two DNS requests to DNS server A and then distributes one DNS request to DNS server B. This algorithm applies to scenarios where DNS servers have different performance and bear similar load for each session.
- **Bandwidth algorithm**—Distributes DNS requests to DNS servers according to the weights and remaining bandwidths of DNS servers. When the remaining bandwidths of two DNS servers are the same, this algorithm is equivalent to the round-robin algorithm. When the weights of two DNS servers are the same, this algorithm always distributes DNS requests to the DNS server that has larger remaining bandwidth.
- **Maximum bandwidth algorithm**—Distributes DNS requests always to an idle DNS server that has the largest remaining bandwidth.

Procedure

1. Enter system view.

   system-view

2. Enter DNS server pool view.

   loadbalance dns-server-pool *pool-name*

3. Specify a scheduling algorithm for the DNS server pool.

   predictor hash address { **destination** | **source** | **source-ip-port** } [**mask** *mask-length*] [**prefix** *prefix-length*]

   predictor { **random** | **round-robin** | { **bandwidth** | **max-bandwidth** } [**inbound** | **outbound**] }

   By default, the scheduling algorithm for a DNS server pool is weighted round robin.

4. Specify the number of DNS servers to participate in scheduling.

   selected-server min *min-number* **max** *max-number*

   By default, the DNS servers with the highest priority participate in scheduling.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect the availability of DNS servers in a DNS server pool.

Restrictions and guidelines

The health monitoring configuration in DNS server view takes precedence over the configuration in DNS server pool view.

You can specify an NQA template for health monitoring. For information about NQA templates, see NQA configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter DNS server pool view.

   `loadbalance dns-server-pool` *`pool-name`*

3. Specify a health monitoring method for the DNS server pool.

   `probe` *`template-name`*

   By default, no health monitoring method is specified for a DNS server pool.

4. Specify the health monitoring success criteria for the DNS server pool.

   `success-criteria` { **`all`** | **`at-least`** *`min-number`* }

   By default, health monitoring succeeds only when all the specified health monitoring methods succeed.

Configuring a DNS server

Perform this task to configure an entity on the LB device for processing DNS requests. DNS servers configured on the LB device correspond to DNS servers in ISP networks. A DNS server can belong to multiple DNS server pools. A DNS server pool can contain multiple DNS servers.

DNS server tasks at a glance

To configure a DNS server, perform the following tasks:

1. Creating a DNS server and specifying a DNS server pool
2. Configuring an IP address for a DNS server

   Choose one of the following tasks:

   - Specifying an IP address and port number
   - Enabling the device to automatically obtain the IP address of a DNS server
3. Specifying a VPN instance
4. Associating a link with a DNS server
5. (Optional.) Setting a weight and priority
6. (Optional.) Configuring health monitoring

Creating a DNS server and specifying a DNS server pool

1. Enter system view.

system-view

2. Create a DNS server and enter its view.

   loadbalance dns-server *dns-server-name*

3. (Optional.) Configure a description for the DNS server.

   description *text*

   By default, no description is configured for a DNS server.

4. Specify a DNS server pool for the DNS server.

   dns-server-pool *pool-name*

   By default, a DNS server does not belong to any DNS server pool.

Specifying an IP address and port number

1. Enter system view.

   system-view

2. Enter DNS server view.

   loadbalance dns-server *dns-server-name*

3. Specify an IP address for the DNS server.

   IPv4:

   ip address *ipv4-address*

   IPv6:

   ipv6 address *ipv6-address*

   By default, no IP address is specified for a DNS server.

4. Specify the port number for the DNS server.

   port *port-number*

   By default, the port number of a DNS server is 0. Packets use their own port numbers.

Specifying a VPN instance

1. Enter system view.

   system-view

2. Enter DNS server view.

   loadbalance dns-server *dns-server-name*

3. Specify a VPN instance for the DNS server.

   vpn-instance *vpn-instance-name*

   By default, a DNS server belongs to the public network.

Enabling the device to automatically obtain the IP address of a DNS server

About this task

In scenarios where IP addresses are obtained through PPPoE, an LB device can dynamically obtain the IP address of a DNS server.

Before configuring this task, you must specify the outgoing interface for the link associated with the DNS server. Otherwise, the IP address of the DNS server cannot be obtained.

Procedure

1. Enter system view.

   `system-view`

2. Enter DNS server view.

   `loadbalance dns-server` *`dns-server-name`*

3. Enable the device to automatically obtain the IP address of the DNS server.

   `auto-alloc address`

   By default, the device does not automatically obtain the IP address of a DNS server.

Associating a link with a DNS server

Restrictions and guidelines

A DNS server can be associated with only one link. A link can be associated with multiple DNS servers.

Procedure

1. Enter system view.

   `system-view`

2. Enter DNS server view.

   `loadbalance dns-server` *`dns-server-name`*

3. Associate a link with the DNS server.

   `link` *`link-name`*

   By default, no link is associated with a DNS server.

Setting a weight and priority

About this task

Perform this task to set a weight for the weighted round robin algorithm and bandwidth algorithm of a DNS server, and set the scheduling priority in the DNS server pool for the DNS server.

Procedure

1. Enter system view.

   `system-view`

2. Enter DNS server view.

   `loadbalance dns-server` *`dns-server-name`*

3. Set a weight for the DNS server.

   `weight` *`weight-value`*

   By default, the weight of a DNS server is 100.

4. Set a priority for the DNS server.

   `priority` *`priority`*

   By default, the priority of a DNS server is 4.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect the availability of a DNS server.

Restrictions and guidelines

The health monitoring configuration in DNS server view takes precedence over the configuration in DNS server pool view.

Procedure

1. Enter system view.

   system-view

2. Enter DNS server view.

   loadbalance dns-server *dns-server-name*

3. Specify a health monitoring method for the DNS server.

   probe *template-name*

   By default, no health monitoring method is specified for a DNS server.

4. Specify the health monitoring success criteria for the DNS server.

   success-criteria { **all** | **at-least** *min-number* }

   By default, health monitoring succeeds only when all the specified health monitoring methods succeed.

Configuring a link

A link is a physical link provided by an ISP. You can guide traffic forwarding by specifying an outbound next hop for a link. You can enhance link performance by configuring the maximum bandwidth, health monitoring, bandwidth ratio, and maximum expected bandwidth.

Link tasks at a glance

To configure a link, perform the following tasks:

1. Creating a link
2. Specifying a next hop IP address or an outgoing interface

   Choose one of the following tasks:

   - Specifying an outbound next hop for a link
   - Specifying an outgoing interface for a link
3. (Optional.) Specifying a VPN instance
4. (Optional.) Configuring the maximum bandwidth
5. (Optional.) Configuring health monitoring
6. (Optional.) Setting the bandwidth ratio and maximum expected bandwidth

Creating a link

1. Enter system view.

   system-view

2. Create a link and enter link view.

   loadbalance link *link-name*

3. (Optional.) Configure a description for the link.

   description *text*

   By default, no description is configured for a link.

Specifying an outbound next hop for a link

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Specify an outbound next hop for the link.

   IPv4:

   router ip *ipv4-address*

   IPv6:

   router ipv6 *ipv6-address*

   By default, no outbound next hop is specified for a link.

Specifying an outgoing interface for a link

About this task

In scenarios where IP addresses are obtained through PPPoE, an LB device can dynamically obtain the outbound next hop IP address through the specified outgoing interface.

Procedure

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Specify an outgoing interface for the link.

   router interface *interface-type interface-number*

   By default, no outgoing interface is specified for a link.

Specifying a VPN instance

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Specify a VPN instance for the link.

   vpn-instance *vpn-instance-name*

   By default, a link belongs to the public network.

Configuring the maximum bandwidth

1. Enter system view.

   system-view

2. Enter link view.

   loadbalance link *link-name*

3. Set the maximum bandwidth for the link.

rate-limit bandwidth [**inbound** | **outbound**] *bandwidth-value* **kbps**

By default, the maximum bandwidth for a link is not limited.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect the availability of a link.

Procedure

1. Enter system view.

   system-view
2. Enter link view.

   loadbalance link *link-name*
3. Specify a health monitoring method for the link.

   probe *template-name*

   By default, no health monitoring method is specified for a link.
4. Specify the health monitoring success criteria for the link.

   success-criteria { **all** | **at-least** *min-number* }

   By default, the health monitoring succeeds only when all the specified health monitoring methods succeed.

Setting the bandwidth ratio and maximum expected bandwidth

About this task

When the traffic exceeds the maximum expected bandwidth multiplied by the bandwidth ratio of a link, new traffic (traffic that does not match any sticky entries) is not distributed to the link. When the traffic drops below the maximum expected bandwidth multiplied by the bandwidth recovery ratio of the link, the link participates in scheduling again.

In addition to being used for link protection, the maximum expected bandwidth is used for remaining bandwidth calculation in the bandwidth algorithm and maximum bandwidth algorithm.

Procedure

1. Enter system view.

   system-view
2. Enter link view.

   loadbalance link *link-name*
3. Set the bandwidth ratio.

   bandwidth [**inbound** | **outbound**] **busy-rate** *busy-rate-number* [**recovery** *recovery-rate-number*]

   By default, the total bandwidth ratio is 70.
4. Set the maximum expected bandwidth.

   max-bandwidth [**inbound** | **outbound**] *bandwidth-value* **kbps**

   By default, the maximum expected bandwidth is not limited.

Configuring an LB class

An LB class classifies packets by comparing packets against specific rules. Matching packets are further processed by LB actions. You can create a maximum of 65535 rules for an LB class.

LB class tasks at a glance

To configure an LB class, perform the following tasks:

1. Creating an LB class
2. Creating a match rule

   Choose the following tasks as needed:

   - Creating a match rule that references an LB class
   - Creating a source IP address match rule
   - Creating a destination IP address match rule
   - Creating an ACL match rule
   - Creating a domain name match rule

Creating an LB class

1. Enter system view.

   system-view
2. Create a DNS LB class, and enter LB class view.

   loadbalance class *class-name* **type dns** [**match-all** | **match-any**]

   When you create an LB class, you must specify the class type. You can enter an existing LB class view without specifying the class type. If you specify the class type when entering an existing LB class view, the class type must be the one specified when you create the LB class.
3. (Optional.) Configure a description for the LB class.

   description *text*

   By default, no description is configured for an LB class.

Creating a match rule that references an LB class

1. Enter system view.

   system-view
2. Enter LB class view.

   loadbalance class *class-name*
3. Create a match rule that references an LB class.

   match [*match-id*] **class** *class-name*

Creating a source IP address match rule

1. Enter system view.

   system-view
2. Enter LB class view.

   loadbalance class *class-name*

3. Create a source IP address match rule.

   match [*match-id*] **source** { **ip address** *ipv4-address* [*mask-length* | *mask*] | **ipv6 address** *ipv6-address* [*prefix-length*] }

Creating a destination IP address match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a destination IP address match rule.

   match [*match-id*] **destination** { **ip address** *ipv4-address* [*mask-length* | *mask*] | **ipv6 address** *ipv6-address* [*prefix-length*] }

Creating an ACL match rule

Restrictions and guidelines

If the specified ACL does not exist, the ACL match rule does not take effect.

Procedure

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create an ACL match rule.

   match [*match-id*] **acl** [**ipv6**] { *acl-number* | **name** *acl-name* }

   By default, an LB class does not have any match rules.

Creating a domain name match rule

1. Enter system view.

   system-view

2. Enter LB class view.

   loadbalance class *class-name*

3. Create a domain name match rule.

   match [*match-id*] **domain-name** *domain-name*

Configuring an LB action

About LB actions

LB actions include the following modes:

- **Forwarding mode**—Determines whether and how to forward packets. If no forwarding action is specified, packets are dropped.
- **Modification mode**—Modifies packets. To prevent the LB device from dropping the modified packets, the modification action must be used together with a forwarding action.

If you create an LB action without specifying any of the previous action modes, packets are dropped.

Restrictions and guidelines

The following tasks are mutually exclusive:

- Configuring the forwarding mode
- Specifying a DNS server pool for guiding packet forwarding
- Skipping the current transparent DNS proxy

Configuring one task automatically cancels the other task that you have configured.

LB action tasks at a glance

To configure an LB action, perform the following tasks:

1. Creating an LB action
2. (Optional.) Configuring a forwarding LB action
   - Configuring the forwarding mode
   - Specifying a DNS server pool for guiding packet forwarding
   - Skipping the current transparent DNS proxy
   - Matching the next rule upon failure to find a DNS server
   - (Optional.) Matching the next rule when all DNS servers are busy
3. (Optional.) Configuring a modification LB action
   - Configuring the ToS field in IP packets sent to the DNS server

Creating an LB action

1. Enter system view.

   `system-view`

2. Create a DNS LB action and enter LB action view.

   `loadbalance action` *`action-name`* **`type dns`**

   When you create an LB action, you must specify the action type. You can enter an existing LB action view without specifying the action type. If you specify the action type when entering an existing LB action view, the action type must be the one specified when you create the LB action.

3. (Optional.) Configure a description for the LB action.

   `description` *`text`*

   By default, no description is configured for an LB action.

Configuring a forwarding LB action

About this task

Three forwarding LB action types are available:

- **Forward**—Forwards matching packets.
- Specify a DNS server pool for guiding packet forwarding.
- **Skip the current transparent DNS proxy**—Skips the current transparent DNS proxy and match the next transparent DNS proxy or virtual server.

- **Match the next rule upon failure to find a DNS server**—If the device fails to find a DNS server according to the LB action, it matches the packet with the next rule in the LB policy.
- Match the next rule when all DNS servers are busy.

Configuring the forwarding mode

1. Enter system view.

   `system-view`

2. Enter DNS LB action view.

   `loadbalance action` *`action-name`*

3. Configure the forwarding mode.

   `forward all`

   By default, the forwarding mode is to discard packets.

   This command does not apply to SIP virtual servers.

Specifying a DNS server pool for guiding packet forwarding

1. Enter system view.

   `system-view`

2. Enter DNS LB action view.

   `loadbalance action` *`action-name`*

3. Specify a DNS server pool for guiding packet forwarding.

   `dns-server-pool` *`pool-name`* [**`sticky`** *`sticky-name`*]

   By default, no DNS server pool is specified for guiding packet forwarding.

Skipping the current transparent DNS proxy

1. Enter system view.

   `system-view`

2. Enter DNS LB action view.

   `loadbalance action` *`action-name`*

3. Skip the current transparent DNS proxy.

   `skip current-dns-proxy`

   By default, the forwarding mode is to discard packets.

Matching the next rule upon failure to find a DNS server

1. Enter system view.

   `system-view`

2. Enter DNS LB action view.

   `loadbalance action` *`action-name`*

3. Match the next rule upon failure to find a DNS server.

   `fallback-action continue`

   By default, the next rule is not matched (packets are dropped) when no DNS servers are available for an LB action.

Matching the next rule when all DNS servers are busy

1. Enter system view.

   `system-view`

2. Enter LB action view.

   `loadbalance action` *`action-name`*

3. Match the next rule when all DNS servers are busy.

   ```
   busy-action continue
   ```

   By default, the device assigns DNS requests to DNS servers regardless of whether they are busy.

Configuring the ToS field in IP packets sent to the DNS server

1. Enter system view.

   ```
   system-view
   ```

2. Enter DNS LB action view.

   ```
   loadbalance action action-name
   ```

3. Configure the ToS field in IP packets sent to the DNS server.

   ```
   set ip tos tos-number
   ```

   By default, the ToS field in IP packets sent to the DNS server is not changed.

Configuring an LB policy

LB policy tasks at a glance

To configure an LB policy, perform the following tasks:

1. Creating an LB policy
2. Specifying an LB action
3. Specifying the default LB action

Creating an LB policy

1. Enter system view.

   ```
   system-view
   ```

2. Create a DNS LB policy and enter LB action view.

   ```
   loadbalance policy policy-name type dns
   ```

   When you create an LB policy, you must specify the policy type. You can enter an existing LB policy view without specifying the policy type. If you specify the policy type when entering an existing LB policy view, the policy type must be the one specified when you create the LB policy.

3. (Optional.) Configure a description for the LB policy.

   ```
   description text
   ```

   By default, no description is configured for an LB policy.

Specifying an LB action

Restrictions and guidelines

A DNS LB policy can reference only DNS LB classes and DNS LB actions.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DNS LB policy view.

   loadbalance policy *policy-name*

3. Specify an LB action for an LB class.

   class *class-name* [**insert-before** *before-class-name* | **insert-after** [*after-class-name*]] **action** *action-name*

   By default, no LB action is specified for an LB class.

Specifying the default LB action

Restrictions and guidelines

The default LB action takes effect on packets that do not match any LB classes.

A DNS LB policy can reference only a DNS LB action as the default LB action.

Procedure

1. Enter system view.

   system-view

2. Enter DNS LB policy view.

   loadbalance policy *policy-name*

3. Specify the default LB action.

   default-class action *action-name*

   By default, no default LB action is specified.

Configuring a sticky group

A sticky group uses a sticky method to distribute similar sessions to the same DNS server according to sticky entries. The sticky method applies to the first packet of a session. Other packets of the session are distributed to the same DNS server.

Sticky group tasks at a glance

To configure a sticky group, perform the following tasks:

1. Creating a sticky group
2. Configuring the IP sticky method
3. (Optional.) Configuring the timeout time for sticky entries

Creating a sticky group

1. Enter system view.

   system-view

2. Create an address- and port-type sticky group and enter sticky group view.

   sticky-group *group-name* **type address-port**

   When you create a sticky group, you must specify the group type. You can enter an existing sticky group view without specifying the group type. If you specify the group type when entering an existing sticky group view, the group type must be the one specified when you create the sticky group.

3. (Optional.) Configure a description for the sticky group.

   description *text*

By default, no description is configured for a sticky group.

Configuring the IP sticky method

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Configure the IP sticky method.

   IPv4:

   ip [**port**] { **both** | **destination** | **source** } [**mask** *mask-length*]

   IPv6:

   ipv6 [**port**] { **both** | **destination** | **source** } [**prefix** *prefix-length*]

   By default, no IP sticky method is configured.

Configuring the timeout time for sticky entries

1. Enter system view.

   system-view

2. Enter sticky group view.

   sticky-group *group-name*

3. Configure the timeout time for sticky entries.

   timeout *timeout-value*

   By default, the timeout time for sticky entries is 60 seconds.

Enabling load balancing logging

About load balancing logging

For security auditing purposes, enable load balancing logging to record load balancing information. Load balancing logging includes NAT logging and link busy state logging.

NAT logging records NAT session information, including IP address and port translation information and access information.

Link busy state logging records busy states for all links.

Enabling load balancing NAT logging

1. Enter system view.

   system-view

2. Enable load balancing NAT logging.

   loadbalance log enable nat

   By default, load balancing NAT logging is disabled.

Enabling load balancing link busy state logging

1. Enter system view.
   system-view
2. Enable load balancing link busy state logging.
   loadbalance log enable bandwidth-busy
   By default, load balancing link busy state logging is disabled.

Displaying and maintaining transparent DNS proxy

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display DNS server pool information.	**display loadbalance dns-server-pool** [**brief** | **name** *pool-name*]
Display DNS server information.	**display loadbalance dns-server** [**brief** | **name** dns-*server-name*]
Display DNS server pool member information.	**display loadbalance dns-server dns-server-pool** *dns-server-pool-name* [**name** *dns-server-name* **port** *port-number*]
Display DNS server statistics.	**display loadbalance dns-server statistics** [**name** *dns-server-name*] [**slot** *slot-number*]
Display DNS server pool member statistics.	**display loadbalance dns-server statistics dns-server-pool** *dns-server-pool-name* [**name** *dns-server-name* **port** *port-number*] [**slot** *slot-number*]
Display transparent DNS proxy information.	**display loadbalance dns-proxy** [**brief** | **name** *dns-proxy-name*]
Display transparent DNS proxy statistics.	**display loadbalance dns-proxy statistics** [**name** *dns-proxy-name*] [**slot** *slot-number*]
Display link information.	**display loadbalance link** [**brief** | **name** *link-name*]
Display link statistics.	**display loadbalance link statistics** [**name** *link-name*] [**slot** *slot-number*]
Display LB class information.	**display loadbalance class** [**name** *class-name*]
Display LB action information.	**display loadbalance action** [**name** *action-name*]
Display LB policy information.	**display loadbalance policy** [**name** *policy-name*]
Display sticky entry information for transparent DNS proxies.	**display sticky dns-proxy** [**dns-proxy-name** *dns-proxy-name*] [**class** { *class-name* | **default-class** } | **client-addr** { *ipv4-address*

Task	Command
	| *ipv6-address* } | **dns-server-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-pool** *pool-name* | **dns-server-port** *port-number* | **key** *sticky-key*] * [**brief**] **display sticky dns-proxy** [**dns-proxy-name** *dns-proxy-name*] [**class** { *class-name* | **default-class** } | **client-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-pool** *pool-name* | **dns-server-port** *port-number* | **key** *sticky-key*] * [**brief**] [**slot** *slot-number*]
Display sticky entry information for transparent DNS proxies.	**display sticky dns-proxy** [*dns-proxy-name*] [**class** *class-name* | **default-class** | **default-dns-server-pool**] [**slot** *slot-number*]
Display sticky group information.	**display sticky-group** [**name** *group-name*]
Display LB hot backup statistics.	**display loadbalance hot-backup statistics** [**slot** *slot-number*]
Clear DNS server statistics.	**reset loadbalance dns-server statistics** [*dns-server-name*]
Clear DNS server pool member statistics.	**reset loadbalance dns-server statistics dns-server-pool** *dns-server-pool-name* [**name** *dns-server-name* **port** *port-number*]
Clear transparent DNS proxy statistics.	**reset loadbalance dns-proxy statistics** [*dns-proxy-name*]
Clear link statistics.	**reset loadbalance link statistics** [*link-name*]
Clear LB hot backup statistics.	**reset loadbalance hot-backup statistics**
Clear sticky entry information for transparent DNS proxies.	**reset sticky dns-proxy** [**dns-proxy-name** *dns-proxy-name*] [**class** { *class-name* | **default-class** } | **client-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-pool** *pool-name* | **dns-server-port** *port-number* | **key** *sticky-key*] * **reset sticky dns-proxy** [**dns-proxy-name** *dns-proxy-name*] [**class** { *class-name* | **default-class** } | **client-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-addr** { *ipv4-address* | *ipv6-address* } | **dns-server-pool** *pool-name* | **dns-server-port** *port-number* | **key** *sticky-key*] * [**slot** *slot-number*]

Transparent DNS proxy configuration examples

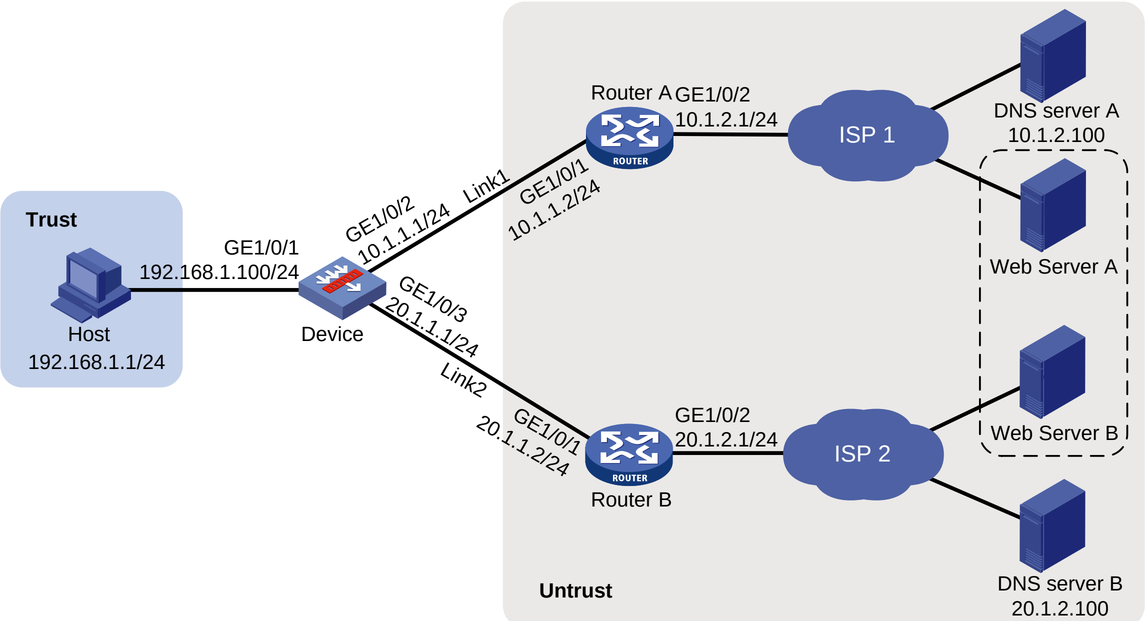
Example: Configuring transparent DNS proxy

Network configuration

In Figure 8, ISP 1 and ISP 2 provide two links with the same bandwidth: Link 1 and Link 2. The IP address of the DNS server of ISP 1 is 10.1.2.100. The IP address of the DNS server of ISP 2 is 20.1.2.100. Intranet users use domain name **www.abc.com** to access Web server A and Web server B.

Configure a transparent DNS proxy on the device to evenly distribute user traffic to Link 1 and Link 2.

Figure 8 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.1.100 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/3
   [Device-security-zone-Untrust] quit
   ```

3. Configure a security policy:

Configure rules to permit traffic from the **Trust** security zone to the **Untrust** security zone and traffic from the **Local** security zone to the **Untrust** security zone, so the users can access the server:

# Configure a rule named **lbrule1** to allow the users to access the server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name lbrule1
[Device-security-policy-ip-1-lbrule1] source-zone trust
[Device-security-policy-ip-1-lbrule1] destination-zone untrust
[Device-security-policy-ip-1-lbrule1] source-ip-subnet 192.168.1.0 255.255.255.0
[Device-security-policy-ip-1-lbrule1] action pass
[Device-security-policy-ip-1-lbrule1] quit
```

# Configure a rule named **lblocalout** to allow the device to send probe packets to the next hop.

```
[Device-security-policy-ip] rule 2 name lblocalout
[Device-security-policy-ip-2-lblocalout] source-zone local
[Device-security-policy-ip-2-lblocalout] destination-zone untrust
[Device-security-policy-ip-2-lblocalout] destination-ip-subnet 10.1.1.0
255.255.255.0
[Device-security-policy-ip-2-lblocalout] destination-ip-subnet 20.1.1.0
255.255.255.0
[Device-security-policy-ip-2-lblocalout] action pass
[Device-security-policy-ip-2-lblocalout] quit
[Device-security-policy-ip] quit
```

4. Configure links:

# Create the link **link1** with next hop address 10.1.1.2.

```
[Device] loadbalance link link1
[Device-lb-link-link1] router ip 10.1.1.2
[Device-lb-link-link1] quit
```

# Create the link **link2** with next hop address 20.1.1.2.

```
[Device] loadbalance link link2
[Device-lb-link-link2] router ip 20.1.1.2
[Device-lb-link-link2] quit
```

5. Create a DNS server pool named **dsp**.

```
[Device] loadbalance dns-server-pool dsp
[Device-lb-dspool-dsp] quit
```

6. Configure DNS servers:

# Create a DNS server named **ds1**, configure its IP address as 10.1.2.100, assign it to DNS server pool **dsp**, and associate it with link **link1**.

```
[Device] loadbalance dns-server ds1
[Device-lb-ds-ds1] ip address 10.1.2.100
[Device-lb-ds-ds1] dns-server-pool dsp
[Device-lb-ds-ds1] link link1
[Device-lb-ds-ds1] quit
```

# Create a DNS server named **ds2**, configure its IP address as 20.1.2.100, assign it to DNS server pool **dsp**, and associate it with link **link2**.

```
[Device] loadbalance dns-server ds2
[Device-lb-ds-ds2] ip address 20.1.2.100
[Device-lb-ds-ds2] dns-server-pool dsp
[Device-lb-ds-ds2] link link2
[Device-lb-ds-ds2] quit
```

7. Configure a transparent DNS proxy:

   # Create a UDP transparent DNS proxy named **dns-proxy1**, configure its IP address as 0.0.0.0, specify DNS server pool **dsp** as its default DNS server pool, and enable the transparent DNS proxy.

```
[Device] loadbalance dns-proxy dns-proxy1 type udp
[Device-lb-dp-udp-dp] ip address 0.0.0.0 0
[Device-lb-dp-udp-dp] default dns-server-pool dsp
[Device-lb-dp-udp-dp] service enable
[Device-lb-dp-udp-dp] quit
```

Verifying the configuration

# Display brief information about all DNS servers.

```
[Device] display loadbalance dns-server brief
DNS server  Address         Port   Link       State      DNS server pool
ds1         10.1.2.100      0      link1      Active     dsp
ds2         20.1.2.100      0      link2      Active     dsp
```

# Display detailed information about all DNS server pools.

```
[Device] display loadbalance dns-server-pool
DNS server pool: dsp
  Description:
  Predictor: Round robin
  Selected server: Disabled
  Probe information:
    Probe success criteria: All
    Probe method:
  Total DNS servers: 2
  Active DNS servers: 2
  DNS server list:
  Name        State         Address         port   Link      Weight   Priority
  ds1         Active        10.1.2.100      0      link1     100      4
  ds2         Active        20.1.2.100      0      link2     100      4
```

# Display detailed information about all transparent DNS proxies.

```
[Device] display loadbalance dns-proxy
DNS proxy: dns-proxy1
  Type: UDP
  State: Active
  Service state: Enabled
  VPN instance:
  IPv4 address: 1.1.1.0/24
  IPv6 address: --
  Port: 53
  DNS server pool: dsp
  Sticky:
  LB policy:
  Connection synchronization: Enabled
  Sticky synchronization: Enabled
  Bandwidth busy protection: Disabled
```

After you complete the previous configuration, the LB device can evenly distribute DNS requests to DNS server A and DNS server B. Then, intranet user traffic is evenly distributed to Link 1 and Link 2.

Configuring inbound link load balancing

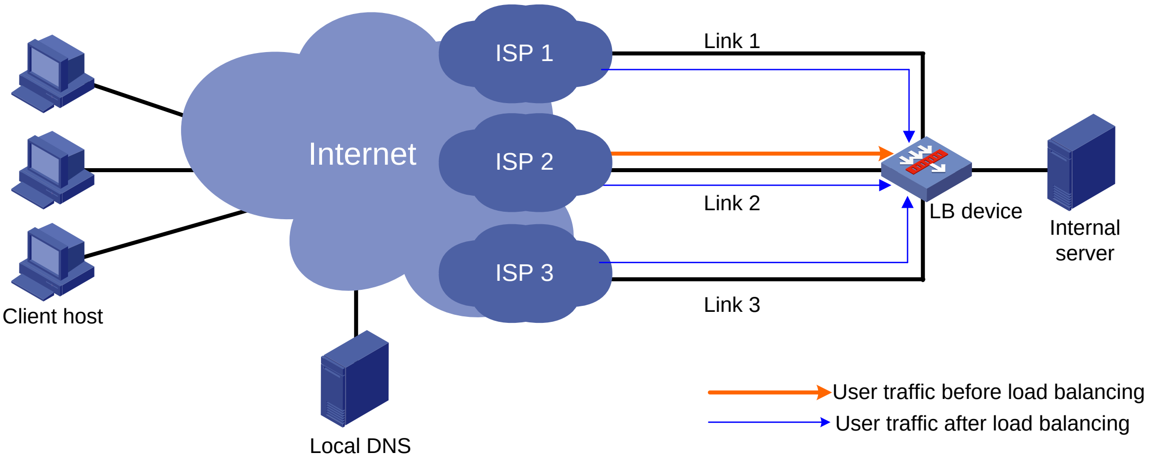
About inbound link load balancing

Inbound link load balancing load balances traffic among the links from the external network to the internal network.

Application scenario

As shown in Figure 9, an enterprise provides services for extranet users through link 1 of ISP 1, link 2 of ISP 2, and link 3 of ISP 3. Inbound link load balancing evenly distributes traffic from extranet users to the internal server on multiple links. This feature can prevent link congestion and implement link switchover upon a link failure.

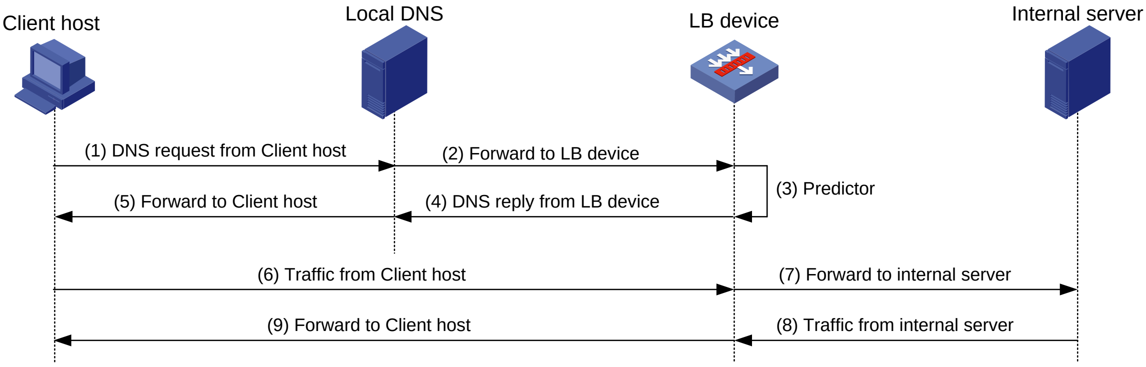
Figure 9 Network diagram



Workflow

Inbound link load balancing is implemented based on DNS resolution. The LB device acts as the authoritative DNS server to process DNS requests from extranet users and select the best link for extranet users. Figure 10 shows the inbound link load balancing workflow.

Figure 10 Inbound link load balancing workflow


Client host
Local DNS
LB device
Internal server
(1) DNS request from Client host
(2) Forward to LB device
(3) Predictor
(5) Forward to Client host
(4) DNS reply from LB device
(6) Traffic from Client host
(7) Forward to internal server
(9) Forward to Client host
(8) Traffic from internal server

Inbound link load balancing uses the following procedure:

1. The client host sends a DNS request to the local DNS server.
2. The local DNS server forwards the DNS request to the LB device.
3. Inbound link load balancing selects a virtual server on the optimal link by using scheduling algorithms, bandwidth limit, and health monitoring method.
4. The LB device sends the virtual server address to the local DNS server in a DNS response.
5. The local DNS server sends the virtual server address to the client host.
6. The client host initiates a connection request to the virtual server address. (The request is forwarded to the LB device.)
7. The LB device initiates a connection request to the internal server.
8. The internal server responds to the LB device.
9. The LB device responds to the client host.

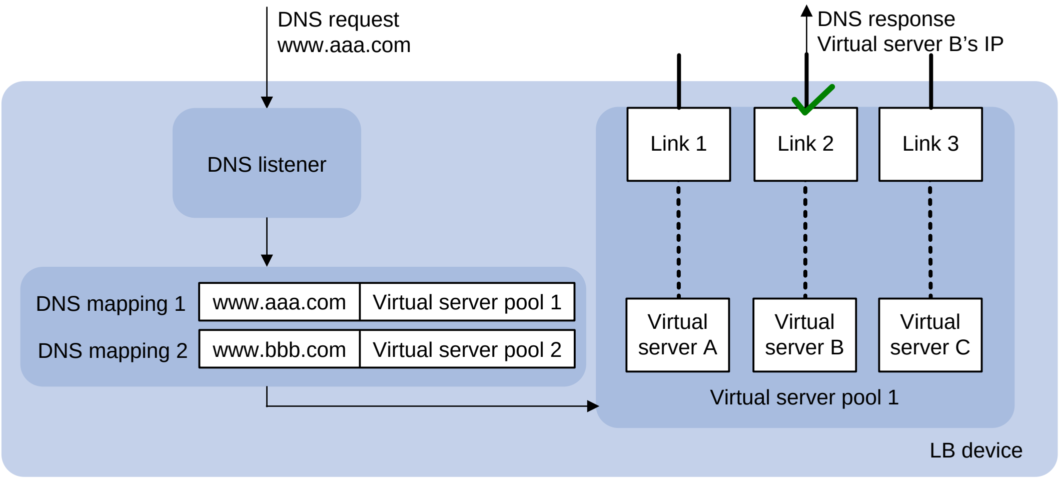
Inbound link load balancing on the LB device

The LB device implements inbound link load balancing by receiving DNS requests and replying with DNS responses that carry IP addresses of virtual severs.

As shown in Figure 11, the LB device contains the following elements:

- **DNS listener**—Listens to DNS requests. When the destination IP address of a DNS request matches that of the DNS listener, the DNS request is processed by inbound link load balancing.
- **DNS mapping**—Associates a domain name with a virtual server pool. The LB device looks up the DNS mappings for the virtual server pool associated with a domain name.
- **Link**—Physical link provided by an ISP.
- **Virtual server pool**—Associates virtual servers with links. The availability of a virtual server and its associated link determines whether the virtual server participates in scheduling.
- **Virtual server**—A virtual entity for processing user services.

Figure 11 Inbound link load balancing on the LB device



If the destination IP address of an incoming DNS request matches that of a DNS listener, the LB device processes the DNS request as follows:

1. The LB device looks up the DNS mappings for the virtual server pool associated with the domain name in the DNS request.
2. The LB device selects the virtual server associated with the best link according to the scheduling algorithm configured for the virtual server pool.

3. The LB device sends the IP address of the selected virtual server in a DNS response to the extranet user.

   The extranet user uses the IP address of the virtual server as the destination IP address and accesses the internal server through the link associated with the virtual server.

Restrictions and guidelines: Inbound link load balancing configuration

You must contact the ISP to configure a delegating domain on the local DNS server to specify the LB device as the authoritative DNS server.

Inbound link load balancing tasks at a glance

To configure inbound link load balancing, perform the following tasks:

1. Configuring a DNS listener
2. Configuring a DNS mapping
3. Configuring a virtual server
4. Configuring a virtual server pool
5. Configuring an LB link
6. (Optional.) Configuring a DNS zone
   - Configuring a DNS forward zone
   - Configuring a DNS reverse zone
7. (Optional.) Configuring a topology
8. (Optional.) Configuring a region
9. (Optional.) Configuring ISP information
10. (Optional.) Enabling load balancing link busy state logging
11. (Optional.) Performing a load balancing test
12. (Optional.) Configuring DNS request parse failure settings
    - Setting the maximum number of DNS request parse failures to be recorded
    - Configuring the types of DNS request parse failures to be recorded

Configuring a DNS listener

DNS listener tasks at a glance

To configure a DNS listener, perform the following tasks:

1. Creating a DNS listener
2. Specifying an IP address and a port number for a DNS listener
3. (Optional.) Specifying a VPN instance
4. Enabling the DNS listening feature
5. (Optional.) Specifying the processing method for DNS mapping search failure

Creating a DNS listener

1. Enter system view.

   system-view

2. Create a DNS listener and enter DNS listener view.

   loadbalance dns-listener *dns-listener-name*

Specifying an IP address and a port number for a DNS listener

About this task

Perform this task to specify an IP address and a port number for the LB device to provide DNS services.

Procedure

1. Enter system view.

   system-view

2. Enter DNS listener view.

   loadbalance dns-listener *dns-listener-name*

3. Specify an IP address and a port number for the DNS listener.

   IPv4:

   ip address *ipv4-address* [**port** *port-number*]

   IPv6:

   ipv6 address *ipv6-address* [**port** *port-number*]

   By default, a DNS listener does not have an IP address or port number.

Specifying a VPN instance

1. Enter system view.

   system-view

2. Enter DNS listener view.

   loadbalance dns-listener *dns-listener-name*

3. Specify a VPN instance for the DNS listener.

   vpn-instance *vpn-instance-name*

   By default, a DNS listener belongs to the public network.

Enabling the DNS listening feature

1. Enter system view.

   system-view

2. Enter DNS listener view.

   loadbalance dns-listener *dns-listener-name*

3. Enable the DNS listening feature.

   service enable

   By default, the DNS listening feature is disabled.

Specifying the processing method for DNS mapping search failure

1. Enter system view.
   system-view
2. Enter DNS listener view.
   loadbalance dns-listener *dns-listener-name*
3. Specify the processing method for DNS mapping search failure.
   fallback { **dns-proxy** | **no-response** | **reject** }
   By default, the processing method is **reject**.

Configuring a DNS mapping

By configuring a DNS mapping, you can associate a domain name with a virtual server pool.

DNS mapping tasks at a glance

To configure a DNS mapping, perform the following tasks:

1. Creating a DNS mapping
2. Specifying a domain name for a DNS mapping
3. Specifying a virtual server pool for a DNS mapping
4. (Optional.) Setting the TTL for DNS records
5. Enabling the DNS mapping feature

Creating a DNS mapping

1. Enter system view.
   system-view
2. Create a DNS mapping and enter DNS mapping view.
   loadbalance dns-map *dns-map-name*

Specifying a domain name for a DNS mapping

Restrictions and guidelines

You can specify multiple domains names for a DNS mapping.

Procedure

1. Enter system view.
   system-view
2. Enter DNS mapping view.
   loadbalance dns-map *dns-map-name*
3. Specify a domain name for the DNS mapping.
   domain-name *domain-name*
   By default, a DNS mapping does not contain domain names.

Specifying a virtual server pool for a DNS mapping

1. Enter system view.
   `system-view`
2. Enter DNS mapping view.
   `loadbalance dns-map` *`dns-map-name`*
3. Specify a virtual server pool for the DNS mapping.
   `virtual-server-pool` *`pool-name`*
   By default, no virtual server pool is specified for a DNS mapping.

Setting the TTL for DNS records

About this task

Perform this task to set a proper TTL to cache DNS records for DNS responses.

- For the DNS client to get the updated DNS record when the LB policy or virtual server configuration changes, set a smaller TTL value, for example, 60 seconds.
- For stable, fast domain name resolution when the network is stable, set a larger TTL value, for example, 86400 seconds.

Procedure

1. Enter system view.
   `system-view`
2. Enter DNS mapping view.
   `loadbalance dns-map` *`dns-map-name`*
3. Set the TTL for DNS records.
   `ttl` *`ttl-value`*
   By default, the TTL for DNS records is 3600 seconds.

Enabling the DNS mapping feature

1. Enter system view.
   `system-view`
2. Enter DNS mapping view.
   `loadbalance dns-map` *`dns-map-name`*
3. Enable the DNS mapping feature.
   `slow-shutdown enable`
   By default, the DNS mapping feature is disabled.

Configuring a virtual server

About this task

Perform this task to specify the IP address and port number that the internal server uses to provide services. Configure inbound link load balancing virtual servers in the same way server load balancing virtual servers are configured. For more information about virtual server configuration

Restrictions and guidelines

- To ensure correct operation of inbound link load balancing when server load balancing is also enabled, do not specify the virtual server's IP address as the DNS listener's IP address. For more information about specifying an IP address for a DNS listener, see "Specifying an IP address and a port number for a DNS listener."
- The virtual server's IPv4 address for inbound link load balancing must be a unicast address with a 32-bit mask length. The IPv4 address cannot be an all-zero address.
- The virtual server's IPv6 address for inbound link load balancing must be a unicast address with a 128-bit prefix length. The IPv6 address cannot be an all-zero address.

Configuring a virtual server pool

Perform this task to facilitate management of virtual servers with similar functions.

Virtual server pool tasks at a glance

To configure a virtual server pool, perform the following tasks:

1. Creating a virtual server pool
2. Adding a virtual server or virtual IP address. Choose the options to configure as needed:
   - Adding a virtual server
   - Adding a virtual IP address
3. (Optional.) Specifying scheduling algorithms for a virtual server pool
4. (Optional.) Enabling the link protection feature

Creating a virtual server pool

1. Enter system view.
   system-view
2. Create a virtual server pool enter virtual server pool view.
   loadbalance virtual-server-pool *name*

Adding a virtual server

1. Enter system view.
   system-view
2. Enter virtual server pool view.
   loadbalance virtual-server-pool *name*
3. Add a virtual server to the virtual server pool.
   virtual-server *virtual-server-name* **link** *link-name* [**weight** *weight-name*]
   By default, no virtual servers are added to a virtual server pool.

Adding a virtual IP address

1. Enter system view.
   system-view

2. Enter virtual server pool view.

   loadbalance virtual-server-pool *name*

3. Add a virtual IP address to the virtual server pool.

   IPv4:

   virtual-ip *ipv4-address* **link** *link-name* [**weight** *weight-value*]

   IPv6:

   virtual-ipv6 *ipv6-address* **link** *link-name* [**weight** *weight-value*]

   By default, no virtual IP addresses are added to a virtual server pool.

Specifying scheduling algorithms for a virtual server pool

About this task

The device provides the following scheduling algorithms for a virtual server pool:

- **Weighted least connection algorithm (least-connection)**—Always assigns DNS requests to the virtual server with the fewest number of weighted active connections (the number of active connections divided by weight).
- **Random algorithm (random)**—Randomly assigns DNS requests to virtual servers.
- **Round robin algorithm (round-robin)**—Assigns DNS requests to virtual servers based on the weights of virtual servers. A higher weight indicates more DNS requests will be assigned.
- **Static proximity algorithm (topology)**—Assigns DNS requests to virtual servers based on static proximity entries.
- **Dynamic proximity algorithm (proximity)**—Assigns DNS requests to virtual servers based on dynamic proximity entries.
- **Bandwidth algorithm (bandwidth)**—Distributes DNS requests to DNS servers according to the weights and remaining bandwidth of DNS servers.
- **Maximum bandwidth algorithm (max-bandwidth)**—Distributes DNS requests always to an idle DNS server that has the largest remaining bandwidth.
- **Source IP address hash algorithm (hash address source)**—Hashes the source IP address of DNS requests and distributes DNS requests to different DNS servers according to the hash values.
- **Source IP address and port hash algorithm (hash address source-ip-port)**—Hashes the source IP address and port number of DNS requests and distributes DNS requests to different DNS servers according to the hash values.
- **Destination IP address hash algorithm (hash address destination)**—Hashes the destination IP address of DNS requests and distributes DNS requests to different DNS servers according to the hash values.

You can specify one preferred scheduling algorithm, one alternative scheduling algorithm, and one backup scheduling algorithm for a virtual server pool. If no virtual server can be selected by using the preferred scheduling algorithm, the alternative scheduling algorithm is used. If no virtual server can be selected by using the alternative scheduling algorithm, the backup scheduling algorithm is used.

Procedure

1. Enter system view.

   system-view

2. Enter virtual server pool view.

   loadbalance virtual-server-pool *name*

3. Specify a scheduling algorithm for the virtual server pool.

predictor { **alternate** | **fallback** | **preferred** } { **least-connection** | **proximity** | **random** | **round-robin** | **topology** **|** { **bandwidth** | **max-bandwidth** } [**inbound** | **outbound**] | **hash address** { **source** | **source-ip-port** | **destination** } [**mask** *mask-length* **|** **prefix** *prefix-length*] }

By default, the preferred scheduling algorithm for the virtual server pool is **round robin**. No alternative or backup scheduling algorithm is specified.

Enabling the link protection feature

About this task

This feature enables a virtual server pool to select a virtual server based on the link bandwidth ratio. If the bandwidth ratio of a link is exceeded, the virtual server is not selected. For more information about configuring the bandwidth ratio, see "Setting the bandwidth ratio and maximum expected bandwidth."

Procedure

1. Enter system view.

   system-view

2. Enter virtual server pool view.

   loadbalance virtual-server-pool *name*

3. Enable the link protection feature.

   bandwidth busy-protection enable

   By default, the link protection feature is disabled.

Configuring an LB link

Link availability is one of the factors that determines whether a virtual server can participate in scheduling. Link availability depends on health monitoring, maximum expected bandwidth, and bandwidth ratio.

LB link tasks at a glance

To configure an LB link, perform the following tasks:

1. Creating an LB link
2. Specifying the outbound next hop for the LB link
3. (Optional.) Configuring health monitoring
4. (Optional.) Setting the bandwidth ratio and maximum expected bandwidth

Creating an LB link

1. Enter system view.

   system-view

2. Create an LB link and enter LB link view.

   loadbalance link *link-name*

Specifying the outbound next hop for the LB link

About this task

The outbound next hop is the IP address of the peer device on the link. You can perform health monitoring and bandwidth limiting on a link specified by the outbound next hop.

Procedure

1. Enter system view.

   `system-view`

2. Enter LB link view.

   `loadbalance link` *`link-name`*

3. Specify the outbound next hop for the LB link.

   `router ip` *`ipv4-address`*

   By default, the outbound next hop is not specified for the LB link.

Configuring health monitoring

About this task

Perform this task to enable health monitoring to detect link quality and status and to ensure link availability. The health monitoring configuration uses NQA templates. For more information about NQA template configuration, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

You can configure multiple health monitoring methods for an LB link.

Procedure

1. Enter system view.

   `system-view`

2. Enter LB link view.

   `loadbalance link` *`link-name`*

3. Specify a health monitoring method for the LB link.

   `probe` *`template-name`*

   By default, no health monitoring method is specified for an LB link.

4. Specify the health monitoring success criteria for the LB link.

   `success-criteria` { **`all`** | **`at-least`** *`min-number`* }

   By default, the health monitoring succeeds only when all the specified health monitoring methods succeed.

Setting the bandwidth ratio and maximum expected bandwidth

About this task

When the traffic exceeds the maximum expected bandwidth multiplied by the bandwidth ratio of a link, new traffic is not distributed to the link. When the traffic drops below the maximum expected bandwidth multiplied by the bandwidth recovery ratio of the link, the link participates in scheduling again.

In addition to being used for link protection, the maximum expected bandwidth is used for remaining bandwidth calculation in the bandwidth algorithm and maximum bandwidth algorithm.

Procedure

1. Enter system view.

   system-view

2. Enter LB link view.

   loadbalance link *name*

3. Set the bandwidth ratio.

   bandwidth [**inbound** | **outbound**] *busy-rate-number* [**recovery** *recovery-rate-number*]

   By default, the total bandwidth ratio is 70.

4. Set the maximum expected bandwidth.

   max-bandwidth [**inbound** | **outbound**] *bandwidth-value* **kbps**

   By default, the maximum expected bandwidth, maximum inbound expected bandwidth, and maximum outbound expected bandwidth are 0. The bandwidths are not limited.

Configuring a DNS forward zone

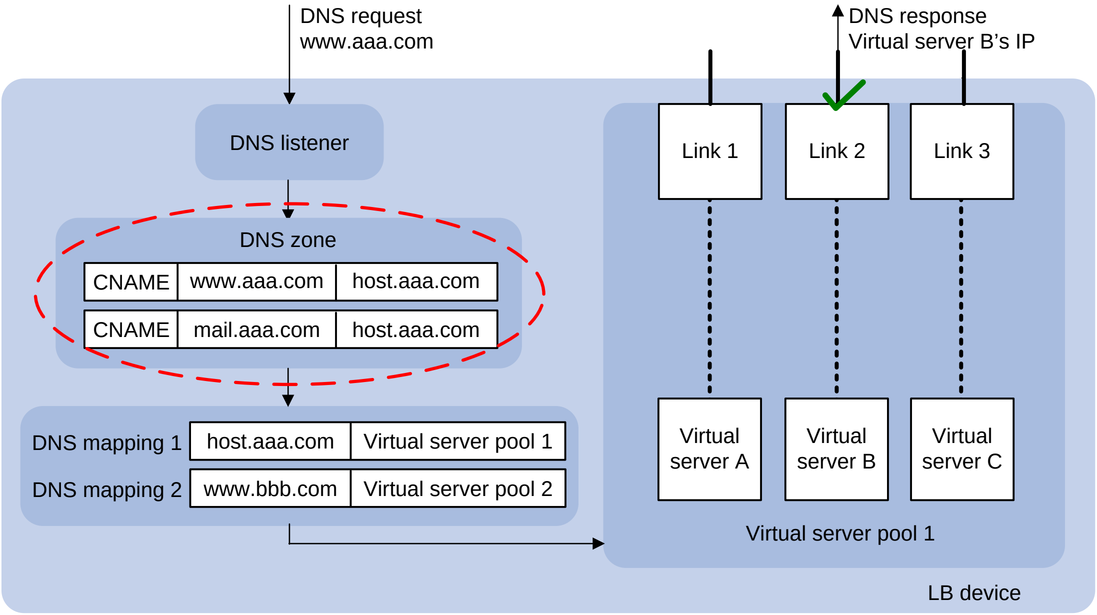
About DNS resource records

During DNS resolution, an LB device looks up the resource records configured in a DNS forward zone for the host name corresponding to the target domain name. DNS resource records are used by an LB device to resolve DNS requests and have the following types:

- **Canonical name (CNAME)**—Maps multiple aliases to one host name (server). For example, an enterprise intranet has a server with host name **host.aaa.com**. The server provides both Web service and mail service. You can configure two aliases (**www.aaa.com** and **mail.aaa.com**) in a CNAME resource record for this server. When a user requests Web service, the user accesses **www.aaa.com**. When a user requests mail service, the user accesses **mail.aaa.com**. Actually, the user accesses **host.aaa.com** in both cases.
- **Mail exchanger (MX)**—Specifies the mail server for a DNS forward zone.
- **Name server (NS)**—Specifies the authoritative DNS server for a DNS forward zone.
- **Start of authority (SOA)**—Specifies authoritative information about a DNS forward zone, including the primary DNS server and administrator mailbox.
- **Service (SRV)**—Specifies the services for a DNS forward zone and the servers that provide these services.
- **Text (TXT)**—Specifies a description for a DNS forward zone.

As shown in Figure 12, the LB device is configured with a DNS forward zone. After receiving a DNS request, the LB device first looks up the resource records in the DNS forward zone for the host name corresponding to the target domain name. Then the LB device looks up the DNS mappings for the IP address of the virtual server associated with the host name.

Figure 12 DNS forward zone workflow on the LB device



DNS forward zone tasks at a glance

To configure a DNS forward zone, perform the following tasks:

1. Creating a DNS forward zone
2. (Optional.) Configuring resource records
   - Configuring a resource record of the specified type

     This task allows you to configure CNAME, MX, NS, SRV, and TXT resource records.
   - (Optional.) Configuring an SOA resource record
3. (Optional.) Setting the TTL for resource records

Creating a DNS forward zone

1. Enter system view.

   `system-view`
2. Create a DNS forward zone and enter DNS forward zone view.

   `loadbalance zone` *`domain-name`*

Configuring a resource record of the specified type

1. Enter system view.

   `system-view`
2. Enter DNS forward zone view.

   `loadbalance zone` *`domain-name`*
3. Configure a resource record of the specified type.

   `record` { **`cname alias`** *`alias-name`* **`canonical`** *`canonical-name`* | **`mx`** [**`host`** *`hostname`*] **`exchanger`** *`exchanger-name`* **`preference`** *`preference`* | **`ns`** [**`sub`** *`subname`*] **`authority`** *`ns-name`* | **`srv`** [**`service`** *`service-name`*]

host-offering-service *hostname* **priority** *priority* **weight** *weight* **port** *port-number* | **txt** [**sub** *subname*] **describe-txt** *description* } [**ttl** *ttl-value*]

By default, a DNS forward zone does not contain resource records.

Configuring an SOA resource record

1. Enter system view.
   system-view
2. Enter DNS forward zone view.
   loadbalance zone *domain-name*
3. Create an SOA resource record and enter SOA view.
   soa
4. Configure the host name for the primary DNS server.
   primary-nameserver *host-name*

   By default, no host name is configured for the primary DNS server.
5. Specify the email address of the administrator.
   responsible-mail *mail-address*

   By default, the email address of the administrator is not specified.
6. Configure the serial number for the DNS forward zone.
   serial *number*

   By default, the serial number for a DNS forward zone is 1.
7. Set the refresh interval.
   refresh *refresh-interval*

   By default, the refresh interval is 3600 seconds.
8. Set the retry interval.
   retry *retry-interval*

   By default, the retry interval is 600 seconds.
9. Set the expiration time.
   expire *expire-time*

   By default, the expiration time is 86400 seconds.
10. Set the minimum TTL.
    min-ttl *ttl-value*

    By default, the minimum TTL is 3600 seconds.

Setting the TTL for resource records

1. Enter system view.
   system-view
2. Enter DNS forward zone view.
   loadbalance zone *domain-name*
3. Set the TTL for resource records.
   ttl *ttl-value*

   By default, the TTL for resource records is 3600 seconds.

Configuring a DNS reverse zone

About this task

The LB device performs reverse DNS resolution according to the DNS reverse zone configuration. Reverse DNS resolution searches for a domain name according to an IP address. The pointer record (PTR) resource records configured in a DNS reverse zone record mappings between domain names and IP addresses.

Reverse DNS resolution is used to address spam attacks by verifying the validity of the email sender. When a mail server receives an email from an external user, it sends a reverse DNS resolution request to the LB device. The LB device resolves the source IP address of the sender into a domain name according to PTR resource records and sends the domain name to the mail server. The mail server compares the received domain name with the actual domain name of the sender. If the two domain names match, the mail server accepts the email. If not, the mail server considers the email as a spam email and discards it.

Procedure

1. Enter system view.

   system-view

2. Create a DNS reverse zone and enter DNS reverse zone view.

   loadbalance reverse-zone { **ip** *ipv4-address mask-length* | **ipv6** *ipv6-address prefix-length* }

3. Configure a PTR resource record.

   record ptr { **ip** *ipv4-address* | **ipv6** *ipv6-address* } *domain-name* [**ttl** *ttl-value*]

   By default, a DNS reverse zone does not contain PTR resource records.

Configuring a topology

About this task

A topology associates the region where the local DNS server resides with the IP address of a virtual server.

When the static proximity algorithm (**topology**) is specified for the virtual server pool, you must configure a topology. For more information about specifying a scheduling algorithm for a virtual server pool, see "Adding a virtual IP address"

When a DNS request matches multiple topology records, the topology record with the highest priority is selected.

Procedure

1. Enter system view.

   system-view

2. Configure a topology.

   topology region *region-name* { **ip** *ipv4-address* { *mask-length* | *mask* } | **ipv6** *ipv6-address prefix-length* } [**priority** *priority*]

Configuring a region

About this task

A region contains network segments corresponding to different ISPs.

Procedure

1. Enter system view.

   system-view

2. Create a region and enter region view.

   loadbalance region *region-name*

3. Add an ISP to the region.

   isp *isp-name*

   By default, a region does not contain any ISPs.

Configuring ISP information

Configure inbound link load balancing ISP information in the same way outbound link load balancing ISP information is configured. For more information about ISP information configuration, see "Configuring outbound link load balancing."

Enabling load balancing link busy state logging

About this task

Perform this task to record busy states for all links.

Procedure

1. Enter system view.

   system-view

2. Enable load balancing link busy state logging.

   loadbalance log enable bandwidth-busy

   By default, load balancing link busy state logging is disabled.

Performing a load balancing test

About performing a load balancing test

Perform this task in any view to test the load balancing result.

Performing an IPv4 load balancing test

To perform an IPv4 load balancing test, execute the following command in any view:

loadbalance local-dns-server schedule-test ip [**vpn-instance** *vpn-instance-name*] **destination** *destination-address* [**destination-port** *destination-port*] **source** *source-address* **source-port** *source-port* **type** { { **a** | **aaaa** | **cname** | **mx** | **ns** | **soa** | **srv** | **txt** } **domain** *domain-name* | **ptr ip address** { *ipv4-address* | *ipv6-address* } } [**slot** *slot-number*]

Performing an IPv6 load balancing test

To perform an IPv6 load balancing test, execute the following command in any view:

loadbalance local-dns-server schedule-test ipv6 [**vpn-instance** *vpn-instance-name*] **destination** *destination-address* [**destination-port** *destination-port*] **source** *source-address* **source-port** *source-port* **type** { { **a** | **aaaa** | **cname** | **mx** | **ns** | **soa** | **srv** | **txt** } **domain** *domain-name* | **ptr ip address** *ipv4-address* } [**slot** *slot-number*]

Setting the maximum number of DNS request parse failures to be recorded

1. Enter system view.

   system-view

2. Set the maximum number of DNS request parse failures that can be recorded.

   loadbalance local-dns-server parse-fail-record max-number *max-number*

   The default setting is 10000.

Configuring the types of DNS request parse failures to be recorded

1. Enter system view.

   system-view

2. Configure the types of DNS request parse failures that can be recorded.

   loadbalance local-dns-server parse-fail-record type { **a** | **aaaa** | **all-disable** | **all-enable** | **cname** | **mx** | **ns** | **ptr** | **soa** | **srv** | **txt** }

   By default, all types of DNS request parse failures are recorded.

Displaying and maintaining inbound link load balancing

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display DNS listener information.	**display loadbalance dns-listener** [**name** *listener-name*]
Display DNS listener statistics.	**display loadbalance dns-listener statistics** [**name** *dns-listener-name*] [**slot** *slot-number*]
Display DNS mapping information.	**display loadbalance dns-map** [**name** *dns-map-name*]
Display DNS mapping statistics.	**display loadbalance dns-map statistics** [**name** *dns-map-name*] [**slot** *slot-number*]
Display virtual server pool information.	**display loadbalance virtual-server-pool** [**brief** | **name** *pool-name*]

Task	Command
Display LB link information.	**display loadbalance link** [**brief** | **name** *link-name*]
Display DNS forward zone information.	**display loadbalance zone** [**name** *domain-name*]
Display DNS reverse zone information.	**display loadbalance reverse-zone** { **ip** [*ipv4-address mask-length*] | **ipv6** [*ipv6-address prefix-length*] }
Display ISP information.	**display loadbalance isp** [**ip** *ipv4-address* | **ipv6** *ipv6-address* | **name** *isp-name*]
Display DNS request parse failures.	**display loadbalance local-dns-server parse-fail-record** [**type** { **a** | **aaaa** | **cname** | **mx** | **ns** | **soa** | **srv** | **txt** }] [**domain** *domain-name*] | **ptr** [**ip address** { *ipv4-address* | *ipv6-address* }]] [**vpn-instance** *vpn-instance-name*] [**slot** *slot-number*]
Clear DNS listener statistics.	**reset loadbalance dns-listener statistics** [*dns-listener-name*]
Clear DNS mapping statistics.	**reset loadbalance dns-map statistics** [*dns-map-name*]
Clear DNS request parse failures.	**reset loadbalance local-dns-server parse-fail-record**

Inbound link load balancing configuration examples

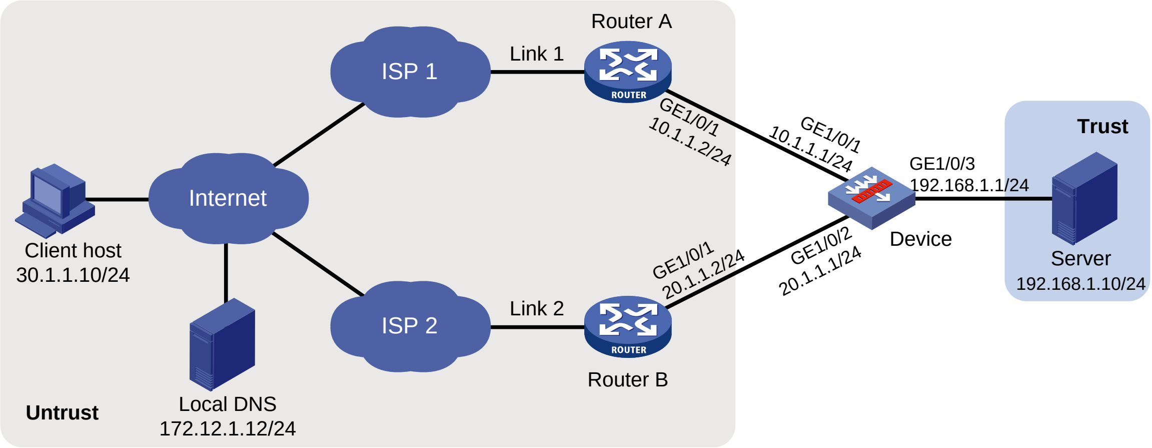
Example: Configuring inbound link load balancing

Network configuration

In Figure 13, ISP 1 and ISP 2 provide two links, Link 1 and Link 2, with the same router hop count, bandwidth, and cost. The internal server uses domain name **l.abc.com** to provide services. The actual host name of the internal server is **www.abc.com**.

Configure inbound link load balancing for the device to select an available link for traffic from the client host to the internal server when a link fails.

Figure 13 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
[Device-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
[Device-security-zone-Untrust] quit
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-security-zone-Trust] quit
```

3. Configure a security policy:

   Configure rules to permit traffic from the **Untrust** security zone to the **Trust** security zone and traffic between the **Untrust** and **Local** security zones, so the users can access the server:

   # Configure a rule named **lbrule1** to allow the users to access the server.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name lbrule1
[Device-security-policy-ip-1-lbrule1] source-zone untrust
[Device-security-policy-ip-1-lbrule1] destination-zone trust
[Device-security-policy-ip-1-lbrule1] destination-ip-subnet 192.168.1.0
255.255.255.0
[Device-security-policy-ip-1-lbrule1] action pass
[Device-security-policy-ip-1-lbrule1] quit
```

   # Configure a rule named **lblocalin** to allow the users to access the DNS listener.

```
[Device-security-policy-ip] rule name lblocalin
[Device-security-policy-ip-2-lblocalout] source-zone untrust
[Device-security-policy-ip-2-lblocalout] destination-zone local
```

```
[Device-security-policy-ip-2-lblocalout] destination-ip-subnet 10.1.1.1
255.255.255.255
[Device-security-policy-ip-2-lblocalout] destination-ip-subnet 20.1.1.1
255.255.255.255
[Device-security-policy-ip-2-lblocalout] action pass
[Device-security-policy-ip-2-lblocalout] quit
[Device-security-policy-ip] quit
```

# Configure a rule named **lblocalout** to allow the device to send probe packets to the next hop.

```
[Device-security-policy-ip] rule name lblocalout
[Device-security-policy-ip-3-lblocalout] source-zone local
[Device-security-policy-ip-3-lblocalout] destination-zone untrust
[Device-security-policy-ip-3-lblocalout] destination-ip-subnet 10.1.1.0
255.255.255.0
[Device-security-policy-ip-3-lblocalout] destination-ip-subnet 20.1.1.0
255.255.255.0
[Device-security-policy-ip-3-lblocalout] action pass
[Device-security-policy-ip-3-lblocalout] quit
[Device-security-policy-ip] quit
```

4. Configure LB links:

# Create the ICMP-type NQA template **t1**.

```
[Device] nqa template icmp t1
[Device-nqatplt-icmp-t1] quit
```

# Create the LB link **link1**, and specify the outbound next hop as 10.1.1.2 and health monitoring method as **t1** for the LB link.

```
[Device] loadbalance link link1
[Device-lb-link-link1] router ip 10.1.1.2
[Device-lb-link-link1] probe t1
[Device-lb-link-link1] quit
```

# Create the LB link **link2**, and specify the outbound next hop as 20.1.1.2 and health monitoring method as **t1** for the LB link.

```
[Device] loadbalance link link2
[Device-lb-link-link2] router ip 20.1.1.2
[Device-lb-link-link2] probe t1
[Device-lb-link-link2] quit
```

5. Create the server farm **sf**.

```
[Device] server-farm sf
[Device-sfarm-sf] quit
```

6. Create the real server **rs** with the IPv4 address 192.168.1.10, and add it to the server farm **sf**.

```
[Device] real-server rs
[Device-rserver-rs] ip address 192.168.1.10
[Device-rserver-rs] server-farm sf
[Device-rserver-rs] quit
```

7. Configure virtual servers:

# Create the HTTP virtual server **vs1** with the VSIP 10.1.1.3 and port number 80, specify its default master server farm **sf**, and enable the virtual server.

```
[Device] virtual-server vs1 type http
[Device-vs-http-vs1] virtual ip address 10.1.1.3
[Device-vs-http-vs1] port 80
[Device-vs-http-vs1] default server-farm sf
```

```
[Device-vs-http-vs1] service enable
[Device-vs-http-vs1] quit
```

# Create the HTTP virtual server **vs2** with the VSIP 20.1.1.3 and port number 80, specify its default master server farm **sf**, and enable the virtual server.

```
[Device] virtual-server vs2 type http
[Device-vs-http-vs2] virtual ip address 20.1.1.3
[Device-vs-http-vs2] port 80
[Device-vs-http-vs2] default server-farm sf
[Device-vs-http-vs2] service enable
[Device-vs-http-vs2] quit
```

8. Create the virtual server pool **vsp**, and add the virtual servers **vs1** and **vs2** associated with the LB links **link1** and **link2** to the virtual server pool.

```
[Device] loadbalance virtual-server-pool vsp
[Device-lb-vspool-vsp] virtual-server vs1 link link1
[Device-lb-vspool-vsp] virtual-server vs2 link link2
```

9. Configure DNS listeners:

# Create the DNS listener **dl1** with the IP address 10.1.1.1, and enable the DNS listener feature.

```
[Device] loadbalance dns-listener dl1
[Device-lb-dl-dl1] ip address 10.1.1.1
[Device-lb-dl-dl1] service enable
[Device-lb-dl-dl1] quit
```

# Create the DNS listener **dl2** with the IP address 20.1.1.1, and enable the DNS listener feature.

```
[Device] loadbalance dns-listener dl2
[Device-lb-dl-dl2] ip address 20.1.1.1
[Device-lb-dl-dl2] service enable
[Device-lb-dl-dl2] quit
```

10. Create the DNS mapping **dm**, specify the domain name **www.aaa.com** and virtual server pool **vsp** for the DNS mapping, and enable the DNS mapping feature.

```
[Device] loadbalance dns-map dm
[Device-lb-dm-dm] domain-name www.aaa.com
[Device-lb-dm-dm] service enable
[Device-lb-dm-dm] virtual-server-pool vsp
[Device-lb-dm-dm] quit
```

11. Configure a DNS forward zone:

# Create a DNS forward zone with domain name **aaa.com**.

```
[Device] loadbalance zone aaa.com
```

# Configure a CNAME resource record by specifying alias **l.aaa.com** for host name **www.aaa.com**.

```
[Device-lb-zone-abc.com] record cname alias l.aaa.com. canonical www.aaa.com. ttl 600
[Device-lb-zone-abc.com] quit
```

Verifying the configuration

# Display information about all DNS listeners.

```
[Device] display loadbalance dns-listener
DNS listener name: dl1
Service state: Enabled
IPv4 address: 10.1.1.1
```

```
Port: 53
IPv6 address: --
IPv6 Port: 53
Fallback: Reject
VPN instance:
```

# Display information about all DNS mappings.

```
[Device] display loadbalance dns-map
DNS mapping name: dm
  Service state: Enabled
TTL: 3600
  Domain name list: www.aaa.com
  Virtual server pool: vsp
```

# Display information about all DNS forward zones.

```
[Device] display loadbalance zone
  Zone name: aaa.com
    TTL: 3600s
    SOA:
   Record list:
     Type    TTL     RDATA
     CNAME   600s    l.aaa.com. www.aaa.com.
```

# Display brief information about all virtual server pools.

```
[Device] display loadbalance virtual-server-pool brief
Predictor: RR - Round robin, RD - Random, LC - Least connection,
           TOP - Topology, PRO - Proximity
           BW - Bandwidth, MBW - Max bandwidth,
           IBW - Inbound bandwidth, OBW - Outbound bandwidth,
           MIBW - Max inbound bandwidth, MOBW - Max outbound bandwidth,
           HASH(SIP) - Hash address source IP,
           HASH(DIP) - Hash address destination IP,
           HASH(SIP-PORT) - Hash address source IP-port
VSpool           Pre    Alt    Fbk    BWP     Total    Active
vsp              RR     LC            Enabled  0        0
```

# Display detailed information about all virtual server pools.

```
[Device] display loadbalance virtual-server-pool
Virtual-server pool: local_pool
  Predictor:
   Preferred RR
   Alternate --
   Fallback  --
  Bandwidth busy-protection: Disabled
  Total virtual servers: 2
  Active virtual servers: 2
  Virtual server list:
  Name      State     Address       Port     Weight  Link
  vs1       Active    10.1.1.3      80        100     link1
  vs2       Active    20.1.1.3      80        100     link2
```

# Display brief information about all real servers.

```
[Device] display real-server brief
Real server      Address              Port  State      VPN instance    Server farm
rs               192.168.1.10         0     Active                     sf
```

# Display brief information about all LB links.

```
[Device] display loadbalance link brief
link         Router IP          State         VPN instance    Link group
link1        10.1.1.2           Active
link2        20.1.1.2           Probe-failed
```

# Display detailed information about all server farms.

```
[Device] display server-farm
Server farm: sf
  Description:
  Predictor: Round robin
  Proximity: Enabled
  NAT: Enabled
  SNAT pool:
  Failed action: Keep
  Active threshold: Disabled
  Slow-online: Disabled
  Probe information:
    Probe success criteria: All
    Probe method:
    t1
  Selected server: Disabled
  Probe information:
    Probe success criteria: All
    Probe method:
    t1
  Total real server: 1
  Active real server: 1
  Real server list:
  Name             State    VPN instance     Address             Port  Weight Priority
  rs                Active                  192.168.1.10         0     100    4
```

# Display brief information about all virtual servers.

```
[Device] display virtual-server brief
Virtual server   State    Type      VPN instance     Virtual address     Port
vs1              Active   HTTP                       10.1.1.3            80
vs2              Active   HTTP                       20.1.1.3            80
```

After you complete the previous configuration, domain name **l.aaa.com** can be resolved into 10.1.1.1 or 20.1.1.1. The client host can access the internal server through Link 1 or Link 2.

NSFOCUS Firewall Series

NF High Availability Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for high availability features.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

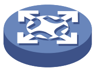
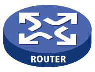
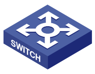
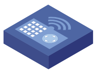
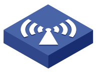
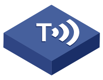
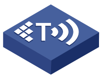
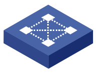
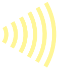
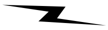
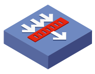
GUI conventions

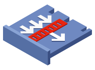
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring hot backup

About hot backup

Hot backup is a device-level HA solution. It enables two devices to back up each other dynamically to ensure user service continuity upon failure of one of the devices.

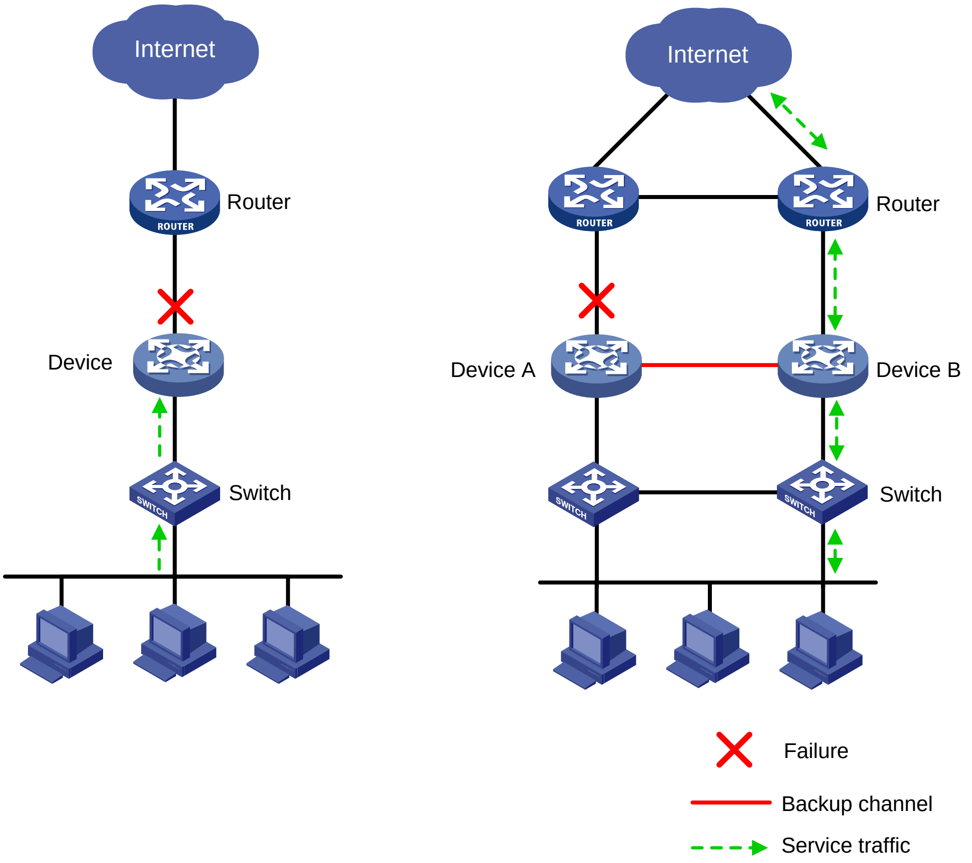
Hot backup works with NSFOCUS proprietary Remote Backup Management (RBM) to manage multiple VRRP groups or adjust the link costs for routing protocols on two member devices to ensure that the devices have consistent roles and states. The hot backup system can synchronize important configuration and service entries between the devices to ensure service continuity. Two devices must have the same software and hardware environments to join a hot backup system.

Application scenario

As shown in Figure 1, typically redundant egress devices are deployed at the border between the external and internal networks to prevent a single point of failure from interrupting traffic forwarding. When one egress device fails, traffic is switched to a different path.

The hot backup system is not used on traditional network devices such as switches and routers as they require only Layer 2 redundancy and route backup to ensure service continuity. It is used on security devices that perform status check and policy-based processing on packets, such as firewalls, IPSs, and network access behavior auditors. These devices check the validity of the first packet of each flow and create a session entry to record the traffic pattern, including the source and destination IP addresses, source and destination ports, and protocol. A security device forwards the subsequent packets of a flow only when the packets match a session entry. To ensure service continuity after traffic is switched between redundant security devices, hot backup is used to synchronize service entries and configuration between the devices through a dedicated channel.

Figure 1 Hot backup network model



Basic hot backup concepts

Basic hot backup concepts are as follows:

- **Primary and secondary roles**—In the control plane, the primary and secondary roles are assigned to the two devices in a hot backup system to control the configuration synchronization between the devices.

  Each hot backup member device adds a prefix to the view prompt to identify its hot backup role. The primary device adds the **RBM_P** prefix, **RBM_P<Sysname>** for example. The secondary device adds the **RBM_S** prefix, **RBM_S<Sysname>** for example.
- **Active and standby states**—Determine which device forwards and processes traffic in the data plane. The active device forwards traffic of services and backs up service entries to the standby device in real time. When the active device fails, the standby device takes over the active role to ensure service continuity.
- **Hot backup channels**—Transmit status information, important configuration, and service entries between the hot backup members.
- **Hot backup modes**—Include active/standby mode and dual-active mode. In active/standby mode, the active device processes all services. In dual-active mode, both devices process services to increase the capability of the hot backup system and load share traffic.
- **Hot backup packets**—RBM packets transmitted through TCP over the hot backup channels between the hot backup members.

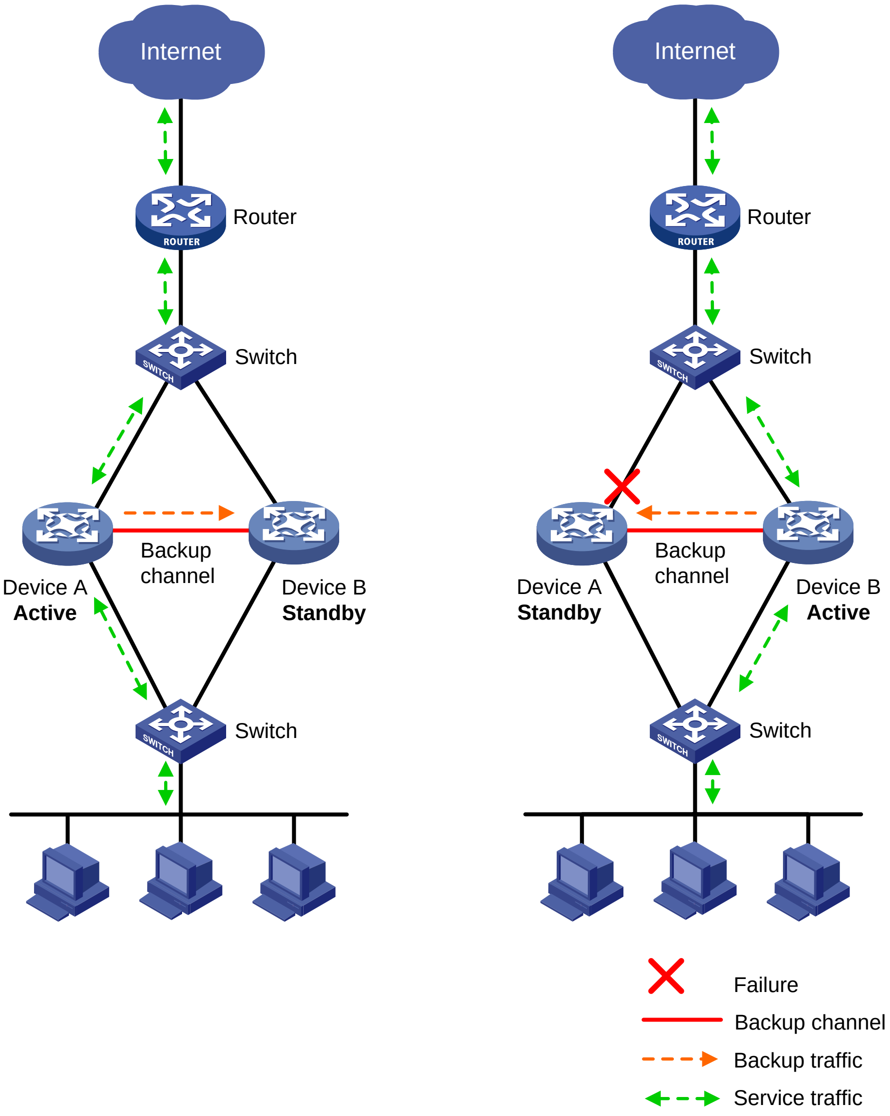
Operating modes of hot backup

Hot backup supports the active/standby and dual-active modes.

Active/standby mode

In active/standby mode, one device is active to process services, and the other device stands by, as shown in Figure 2. When an interface or link on the active device fails or when the active device fails, the standby device becomes active to process services.

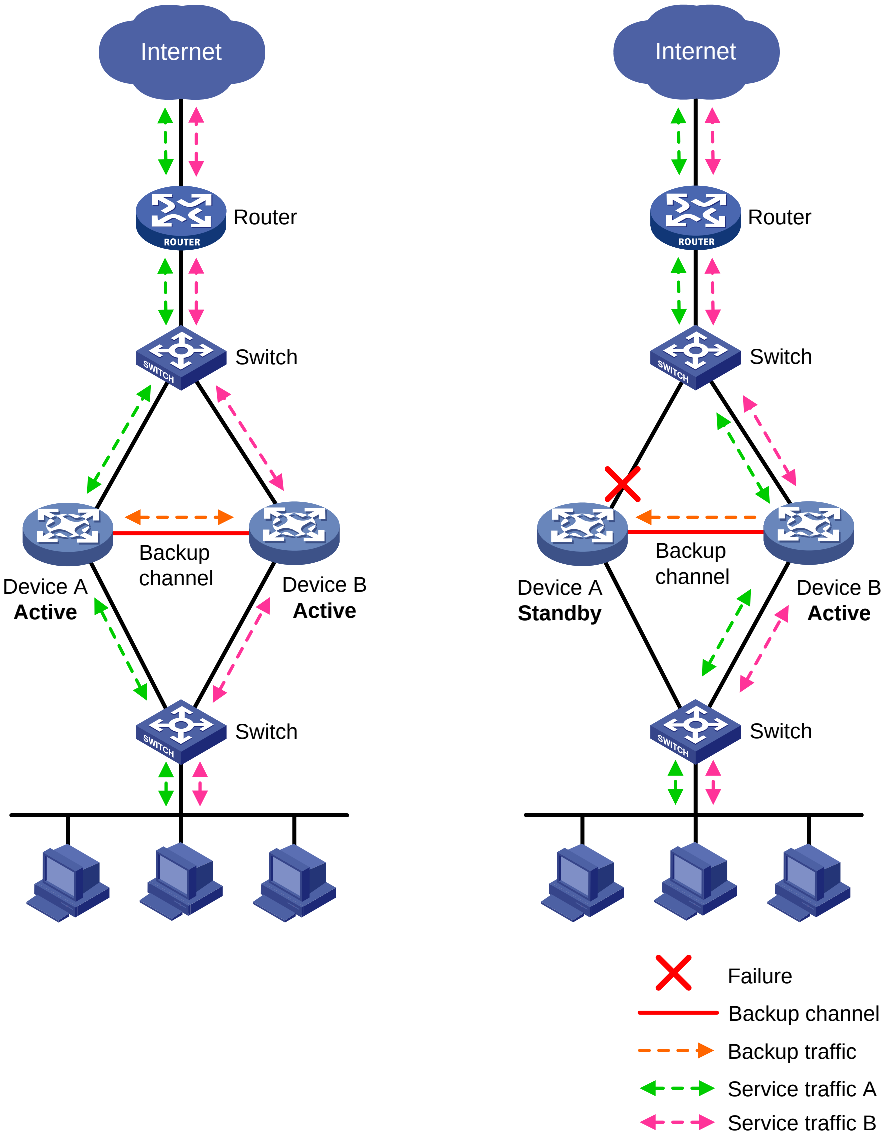
Figure 2 Active/standby mode of hot backup



Dual-active mode

In dual-active mode, both devices process services to increase capability of the hot backup system, as shown in Figure 3. When one device fails, its traffic is switched to the other device for forwarding.

Figure 3 Dual-active mode of hot backup



Hot backup data synchronization

Hot backup packets

Hot backup packets include the following:

- **Keepalive packets**—Transmitted periodically between the hot backup members for peer availability detection.
- **Control packets**—Control active/standby switchovers based on the running status.
- **Backup packets**—Back up configuration and service entries between the hot backup members.
- **Configuration consistency check packets**—Check whether the hot backup members have consistent configuration.
- **Transparent transmission packets**—Transmit or replicate asymmetric-path traffic between the hot backup members.

Hot backup channels

Hot backup transmits hot backup status, important configuration, and service entries between the hot backup members through the following channels:

- **Control channel**—Transmits data by using packets, including hot backup status packets, configuration consistency check packets, and packets used for configuration synchronization.
- **Data channel**—Transmits backup packets and packets that require transparent transmission. The data channel uses the hardware driver for data transmission and supports only Layer 2 forwarding.

The control channel uses the keepalive mechanism of TCP for reachability detection. Each member device periodically sends hot backup keepalive packets to the hot backup peer over the hot backup control channel. If a device has not received any responses from the peer when the maximum number of hot backup keepalive attempts is reached, the hot backup control channel is disconnected.

Service entry backup

Hot backup backs up the service entries generated on the active device to the standby device to prevent service interruption when an active/standby switchover occurs.

Security devices generate a session entry for each dynamic connection. In the hot backup system, only the active device processes traffic and generates session entries. To ensure service continuity, the active device backs up its session entries to the standby device in real time. After an active/standby switchover, the new active device can forward the packets of the existing services based on the session entries without interruption.

Hot backup can perform hot backup for the following service entries:

- NAT port blocks.
- AFT port blocks.
- Session entries.
- Session relation entries.
- Entries generated by security service modules.

Configuration backup

Hot backup backs up important configuration from the primary device to the secondary device to prevent service interruption when an active/standby switchover occurs.

- When both devices are operating correctly, the primary device synchronizes configuration to the secondary device. The configuration on the secondary device is overwritten. As a best practice to ensure correct operation of hot backup, enable configuration backup on the primary device.
- When one of the devices reboots, the device that completes reboot obtains configuration from the device that is not rebooted. The configuration on the rebooted device is overwritten.

Hot backup supports both automatic backup and manual backup.

Hot backup can perform configuration backup for the following services:

- **Resources**—VPN instance, ACL (excluding the **`acl copy`** command), object group, time range, security zone, session management, APR, and AAA (excluding the commands for local users and local guests).
- **DPI**—Application layer inspection engine, IPS, URL filter, data filter, file filter, anti-virus, data analysis center.
- **Polices**—Security policy, ASPF, attack detection and prevention (excluding the **`blacklist ip`** command), connection limit, NAT, AFT, load balancing, bandwidth management, application auditing and management, shared network access management, and proxy policy.
- **Logs**—Fast log output (excluding the **`customlog host source`** command) and flow log (excluding the **`userlog flow export source-ip`** command).

- **VPN**—SSL VPN.
- **Any other services**—VLAN and cloud connection.

Configuration consistency check

Hot backup verifies configuration consistency between the hot backup members by using configuration consistency check packets. If a device detects configuration inconsistency, it generates a log for you to manually synchronize configuration.

Configuration consistency check operates as follows:

1. The primary device sends configuration consistency check packets to the secondary device and collects configuration digests of related modules at the same time.
2. The secondary device receives the packets, encapsulates its configuration digests into configuration consistency check packets, and sends these packets to the primary device.
3. The primary device compares its configuration digests with those of the secondary device. If inconsistency is detected, the primary device generates a log.

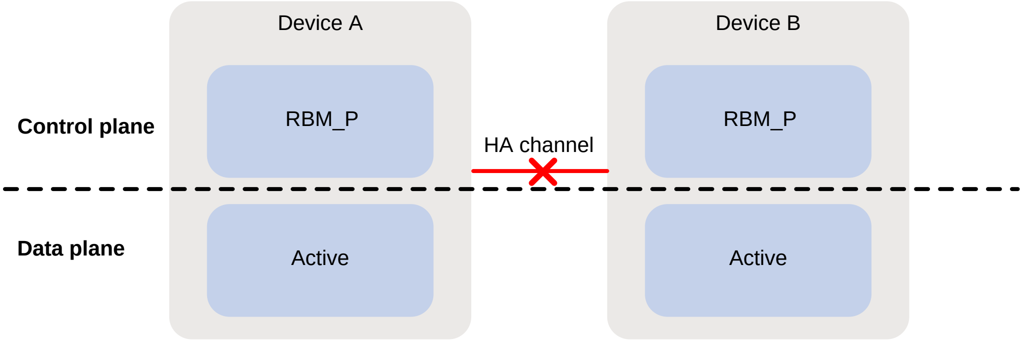
Running status switchover

The hot backup role is determined based on configuration. The running status is determined based on hot backup role election and can be switched over.

Running status switchover process

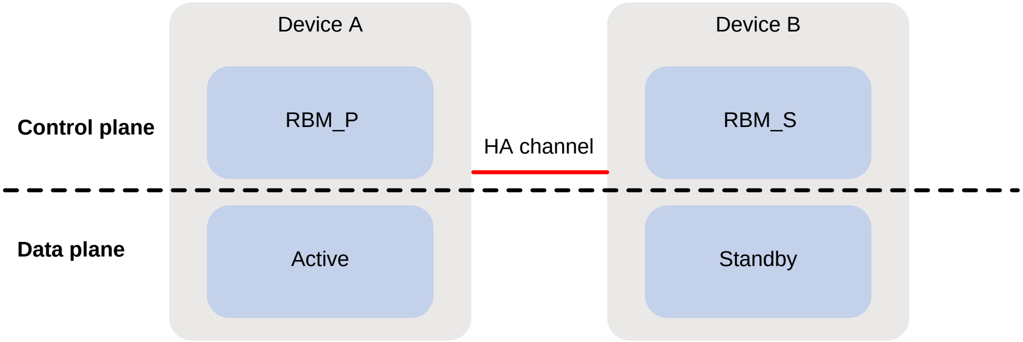
As shown in Figure 4, both devices are primary in the control plane and active in the data plane when the hot backup channels are not established. This is not a correct condition because the hot backup system has not set up.

Figure 4 Hot backup channels not established



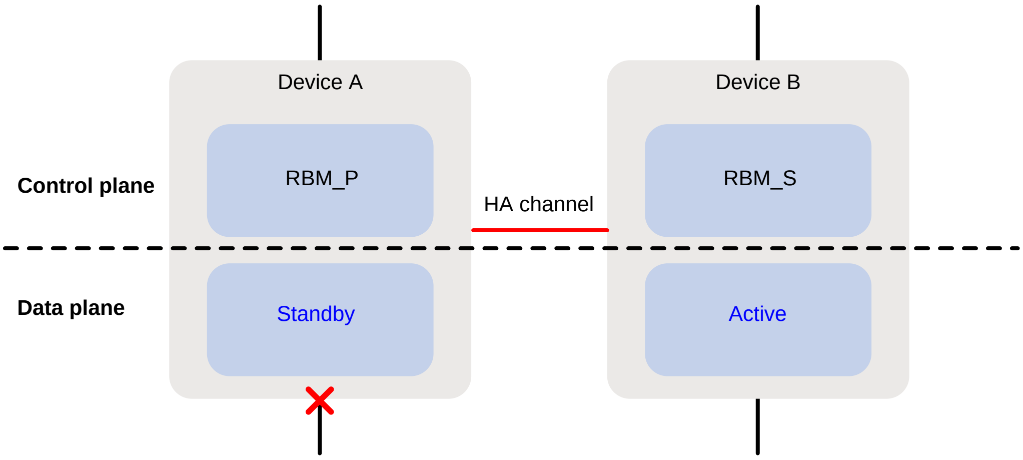
As shown Figure 5, when the hot backup channels are established and both devices are operating correctly, the hot backup roles of the devices are determined based on configuration and do not change in any condition. In the data plane, the running status of the devices is consistent with their hot backup roles. In active/standby mode, the primary device is the active device, and the secondary device is the standby device. In dual-active mode, both devices are active.

Figure 5 Hot backup in active/standby mode



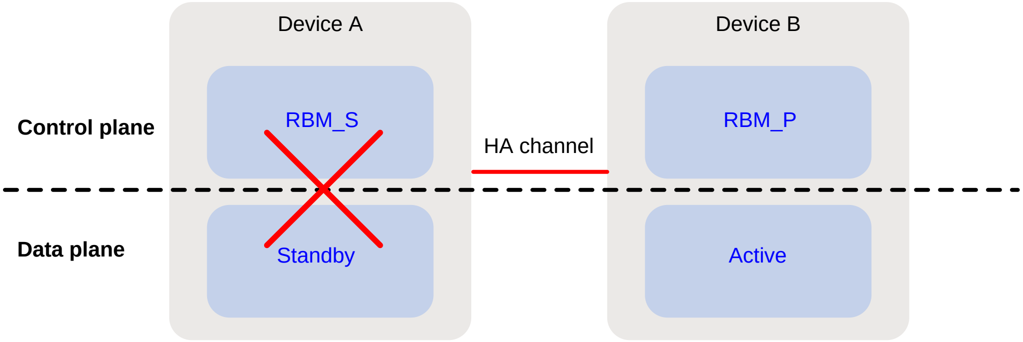
As shown in Figure 6, when the uplink or downlink fails on a device, the status of the devices changes in the data plane. The device that operates correctly processes all traffic. In the control plane, the hot backup roles of the devices do not change.

Figure 6 Uplink or downlink failure



As shown in Figure 7, the hot backup role changes in the control plane only when the primary device fails or restarts, and the running status in the data plane will change accordingly. The secondary device will take over the primary role. When the original primary device recovers, it takes over the primary role.

Figure 7 Device failure or restart



Running status switchover triggers

In the hot backup system, an active/standby switchover occurs if one device fails or its links fail. The other device that operates correctly will take over to process all traffic that traverse the hot backup system to ensure service continuity. The following events can trigger an active/standby switchover.

- Disconnection of the hot backup control channel.

  If the hot backup control channel is disconnected when both member devices are operating correctly, both devices become active devices. However, the devices will leave the hot backup system in this situation, and forwarding of asymmetric-path traffic will be interrupted.
- Failure of the active device.
- Failure of the interface monitored by hot backup on the active device.
- State change to Negative of any track entry monitored by hot backup on the active device.

  This event triggers an active device election. If both member devices have track entries in Negative state, the following rules apply:
  - In active/standby mode, the primary device is the active device, and the secondary device is the standby device.
  - In dual-active mode, both member devices are active.

When an active/standby switchover occurs, hot backup collaborates with other modules to direct traffic to the new active device.

- When working with VRRP, hot backup places all VRRP groups on the failed device to backup state.
- When working with a routing protocol, hot backup increases the link costs of routes on the failed device.

Associating hot backup with VRRP

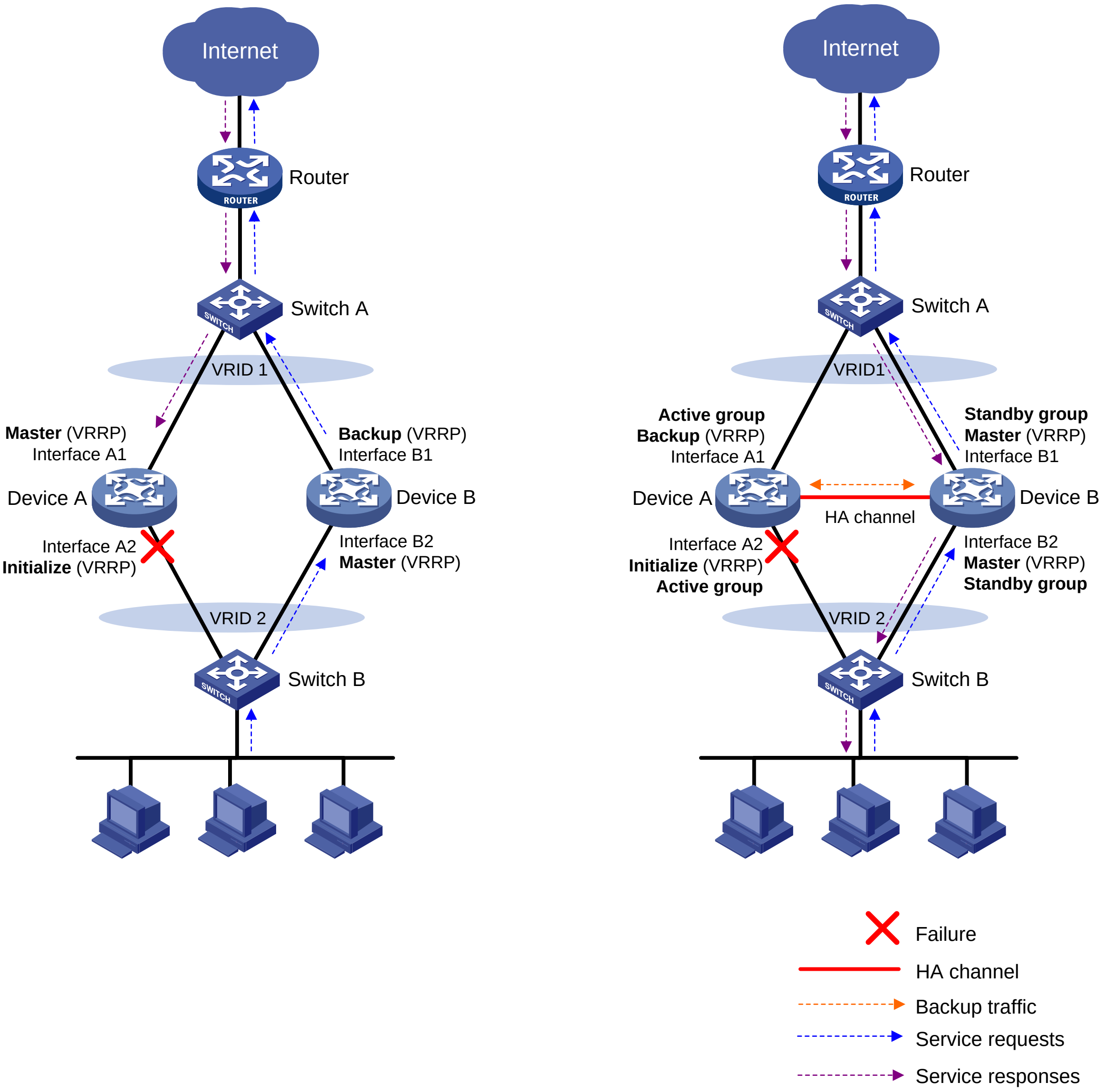
About hot backup and VRRP association

You can use hot backup and VRRP in combination to control master/backup switchover for device role consistency (master or backup) in multiple VRRP groups. This ensures that both inbound and outbound traffic can be switched to the new master for symmetric forwarding upon device failure.

Figure 8 illustrates VRRP association with hot backup in active/standby mode.

- As shown in the left, VRRP cannot ensure symmetric forwarding upon failure on a device, which causes traffic interruption.
- As shown in the right, after the hot backup control channel is established, hot backup determines the roles of the devices in all VRRP groups. The master election mechanism of VRRP no longer takes effect. If the hot backup control channel is disconnected, the master election mechanism of VRRP takes effect again.

Figure 8 VRRP and hot backup association



VRRP active/standby group

Hot backup is associated with VRRP by VRRP active and standby groups.

A VRRP active/standby group can be in master or backup state, which determines the state of devices in the associated VRRP groups. For example, if a VRRP active group is in master state, all devices in the associated VRRP groups are masters.

The initial state of a VRRP active/standby group is depends on the hot backup mode.

- **Active/Standby mode**—On the primary device, the initial state is master for the VRRP active and standby groups. On the secondary device, the initial state is backup for the VRRP active and standby groups.
- **Dual-active mode**—The state of a VRRP active/standby group is not affected by the hot backup roles. The initial state is master for the VRRP active group and is backup for the VRRP standby group.

VRRP master election in the hot backup environment

After hot backup is associated with VRRP, hot backup determines the roles of the devices in the VRRP groups. As shown in Figure 8, Device A is the master in VRRP group 1 and VRRP group 2, and Device B is the backup in VRRP group 1 and VRRP group 2. When Interface A2 on Device A fails, the following events occur:

1. Hot backup receives an interface failure event and sends the status change information of the VRRP active and standby groups to Device B.
2. Device B sets its role to master in the VRRP standby group and then becomes the master in VRRP group 1 and VRRP group 2.
3. Device B sends a response to Device A after the master/backup switchover.
4. Device A sets its role to backup in the VRRP active group and then becomes the backup in VRRP group 1 and VRRP group 2.

When Interface A2 recovers, hot backup performs another master/backup switchover following the same procedure. Traffic is switched back to Device A after the switchover.

ARP and MAC learning in VRRP

When the members of a VRRP group receive an ARP request for the group's virtual IP address, the master replies with the group's virtual MAC address. This allows the upstream and downstream Layer 2 devices and hosts to learn the virtual MAC address.

Associating hot backup with routing protocols

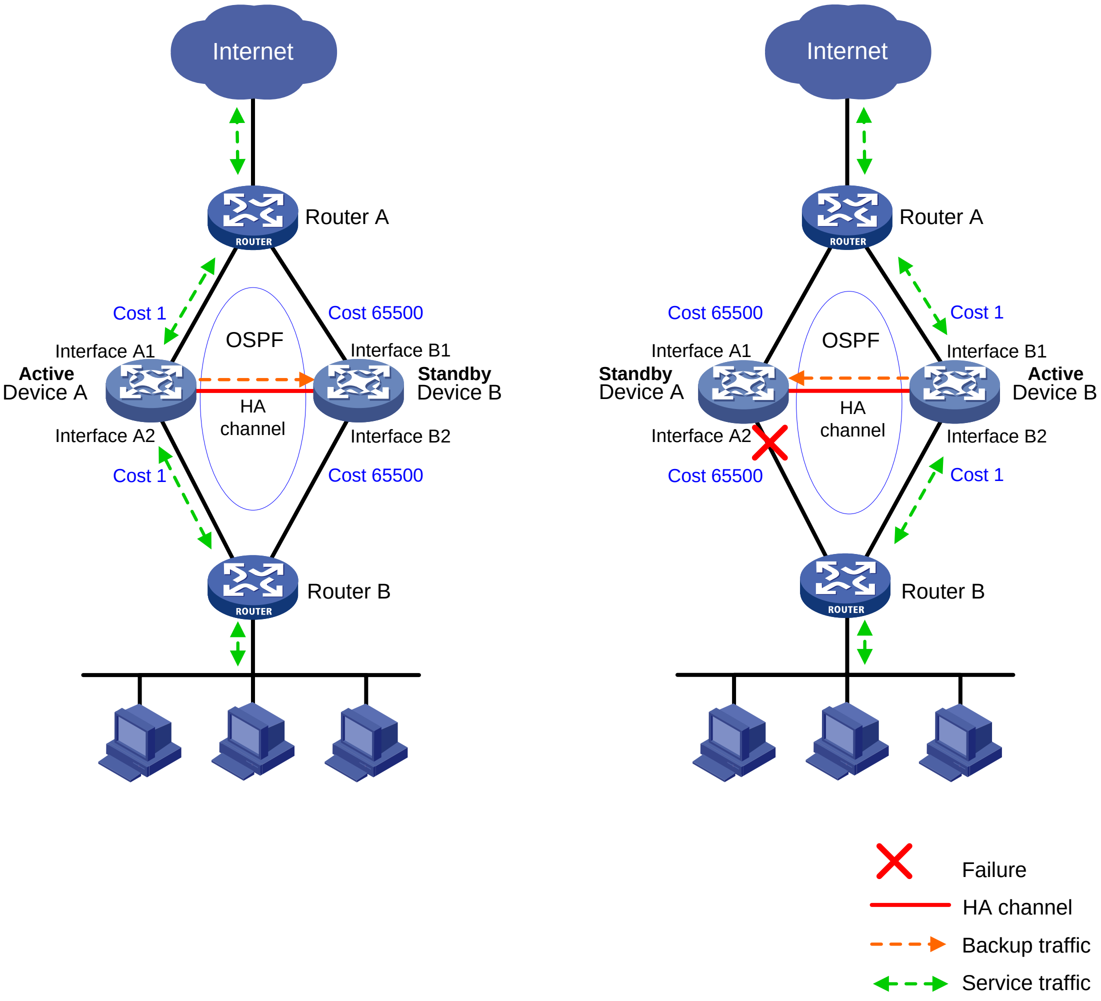
You can use hot backup to enable the routing protocols on the standby device to advertise modified link cost. The feature ensures that both inbound and outbound traffic can be switched to the new active device for symmetric forwarding.

This feature requires associating hot backup with Track to perform active/standby switchover upon link or interface failures.

Figure 9 illustrates OSPF association with hot backup in active/standby mode.

- As shown in the left, Device A (active device) advertises link cost 1 based on OSPF configuration. Device B (standby device) advertises link cost 65500 modified by hot backup. Both inbound and outbound traffic are forwarded through Device A.
- As shown in the right, when interface A2 fails, Device A and Device B perform an active/standby switchover. After the switchover is complete, Device B (active device) advertises link cost 1 based on OSPF configuration. Device A (standby device) advertises link cost 65500 modified by hot backup. Both inbound and outbound traffic are forwarded through Device B.

Figure 9 OSPF and hot backup association



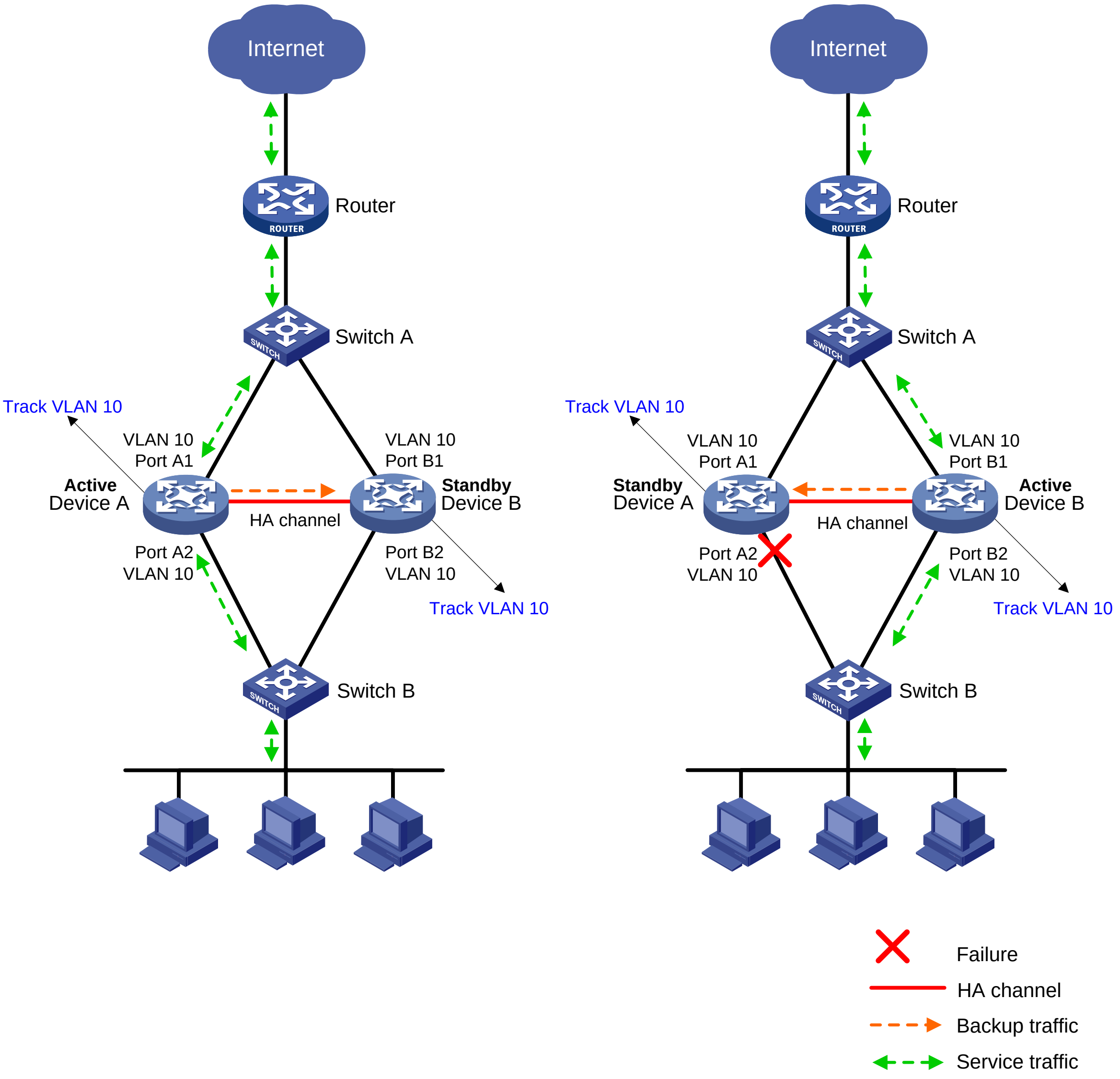
Transparent in-path deployment of hot backup

When you use this networking scheme, you can use the **track vlan** or **track interface** command to enable collaboration between uplink and downlink interfaces. The Track configuration ensures that a group of interfaces have the same status, and uplink and downlink traffic can be switched simultaneously between the member devices.

The following information uses the **track vlan** setting as an example to describe how interfaces collaborate:

- As shown in Figure 10, when both Device A (active) and Device B (standby) are operating correctly, tracked VLAN 10 is in active state on Device A and in inactive state on Device B. As a result, Device A forwards all traffic that traverses the hot backup system.
- As shown in Figure 10, when downlink Port A2 of Device A fails, Device A and Device B switch their roles. Then, hot backup places VLAN 10 in inactive state on Device A (standby) and in active state on Device B (active). As a result, Device B forwards all traffic that traverses the hot backup system.

Figure 10 Transparent in-path deployment of hot backup



Restrictions and guidelines: Hot backup configuration

Member device restrictions and guidelines

A hot backup system can contain a maximum of two devices.

To ensure that the traffic size is within the processing capability of one device upon failure of the other device, make sure the throughput of each device does not exceed 50% of its capability.

Configuration synchronization restrictions and guidelines

For the service modules for which hot backup supports configuration synchronization, you need to configure relevant settings only on the primary device. For the service modules that do not support configuration synchronization, you need to configure relevant settings on all devices in the hot backup system. For support of hot backup for configuration synchronization of service modules, see "Configuration backup."

For resource file associated functions or files (such as public key information, ISP address library file, and signature file), you need import the file with the same content to all devices in the hot backup system.

For the features or signature libraries that require licenses, purchase separate licenses and activate them on the hot backup member devices.

Interface restrictions and guidelines

On each member device, you must assign all interfaces that transmit service traffic to security zones and configure security policies to enable inter-zone communication.

Do not use hot backup in combination with the features that obtain IP addresses automatically, DHCP client for example. The interfaces on the hot backup system must use static IP addresses.

If service interfaces work in Layer 2 mode, assign uplink and downlink service interfaces to the same VLAN.

Hot backup channel restrictions and guidelines

The hot backup channels are set up over the keepalive link that is a direct link or traverses only Layer 2 switches.

When you select interfaces for hot backup channel setup, follow these restrictions and guidelines:

- Use physical interfaces or aggregate interfaces. Do not use subinterfaces or aggregation member ports.
- As a best practice, use the same physical or logical link to convey the hot backup data and control channels. Use the default MTU on the interfaces used for setting up the link.

When configuration backup is in progress, do not perform active/standby switchovers, remove or install service modules, or change the configuration. To view the backup progress, execute the **display remote-backup-group status** command.

If the hot backup system performs asymmetric forwarding, use the **session aging-time state fin** command to set the aging time for TCP sessions in FIN_WAIT state to 15 seconds. This configuration speeds up session entry aging when TCP connections disconnect and thus saves system resources. For more information about the **session aging-time state** command, see session management commands in *Security Command Reference*.

Hot backup deployment restrictions and guidelines

When you deploy hot backup, follow the restrictions and guidelines for the network scheme you have chosen.

Routed in-path hot backup deployment between upstream and downstream routers

You can use either active/standby or dual-active mode.

If you use hot backup with a routing protocol, use the **adjust-cost enable** command to enable hot backup to adjust the link costs for the routing protocol. In addition, associate hot backup with track entries to monitor the status of uplink and downlink interfaces.

If you use hot backup with static routes, use the **track interface** command to monitor uplink and downlink Layer 3 Ethernet interfaces.

Routed in-path hot backup deployment between upstream and downstream switches

You can use either active/standby or dual-active mode.

You must use hot backup with VRRP and configure hot backup to work with track entries to monitor the status of uplink and downlink interfaces.

Transparent in-path hot backup deployment between upstream and downstream routers

You can use either active/standby or dual-active mode.

You must use the **`track interface`** command to enable hot backup to monitor the status of uplink and downlink Layer 2 Ethernet interfaces.

Transparent in-path hot backup deployment between upstream and downstream switches

You can use only active/standby mode.

You must use the **`track vlan`** command to enable hot backup to monitor the status of uplink and downlink VLANs.

Feature compatibility restrictions

Compatibility with contexts

You can configure hot backup only on the default context. The configuration will be issued to all non-default contexts.

When an active/standby switchover occurs on any context, the other contexts will perform an active/standby switchover accordingly. As a best practice, place the active member devices of all contexts on the same device.

If you use hot backup on a non-default context, you must assign service interfaces to the non-default context in shared mode.

Compatibility with NAT

When you use NAT on a hot backup system, follow these restrictions:

- Hot backup does not support detecting reachability of NAT address group members or Easy IP.
- NAT address groups cannot contain the IP addresses of any interfaces on the hot backup member devices. If you violate this restriction, both hot backup member devices will respond to the ARP packets that an upstream device sends to request the IP addresses in the NAT address groups. As a result, ARP conflicts occur.
- The source or destination IP address in a NAT policy does not belong to the interface used for setting up the hot backup channels. Violation of this restriction causes keepalive link faults.
- If you cannot ensure that a flow identified by a five-tuple is processed by the same device in one direction, you must use the **`nat remote-backup port-alloc`** { **`primary`** | **`secondary`** } command to specify NAT port ranges for both hot backup member devices.

When you use NAT on a hot backup system operating in dual-active mode, follow these restrictions:

- NAT address groups do not support EIM.
- If NAT operates in PAT mode, you must execute the following commands on the NAT group members, one command on one device:
  - **`nat remote-backup port-alloc primary`**
  - **`nat remote-backup port-alloc secondary`**

  These commands divide an address group into two halves to avoid conflicts.
- If NAT operates in NO-PAT mode, you must configure two address groups to avoid resource assignment conflicts.

Compatibility with other protocols

Make sure the control and data packets of a multi-channel protocol such as FTP or SIP are processed by the same hot backup member device.

In the hot backup environment, to process traffic based on the SCTP protocol, make sure both the outbound and inbound packets of the same flow are processed by the same device.

Hardware environment consistency

Before you configure hot backup, verify that the following hardware settings are the same on the devices to be assigned to a hot backup system:

- Device model.
- Number and type of management interfaces, service interfaces, and interfaces for setting up the hot backup channels. Do not use one interface for multiple purposes.
- Location, number, and type of disks. A device not with disks installed has small log storage and do not support some types of logs or reports.

Software environment consistency

Before you configure hot backup, verify that the following software settings are the same on the devices to be assigned to a hot backup system:

- Software environment and version, including boot packages, system packages, feature packages, and patches.
- Licensed signature libraries and features, such as signature library types, signature library version, validation time, and number of licensed resources.
- Interface numbers.
- Type, speed, and number of the interfaces for setting up the hot backup channels. As a best practice, use aggregate interfaces.
- Aggregate interface numbers and aggregation member port numbers.
- Security zone configuration on the interfaces at the same location.
- Multi-CPU packet distribution policy (configurable with the **`forwarding policy`** command).

Network interconnection restrictions

Configure security policies for the hot backup members to correctly exchange the packets required in forwarding over the hot backup channels, routing protocol packets for example. As a best practice, make sure only those packets are transmitted between the security zone that accommodates service interfaces and the **Local** security zone.

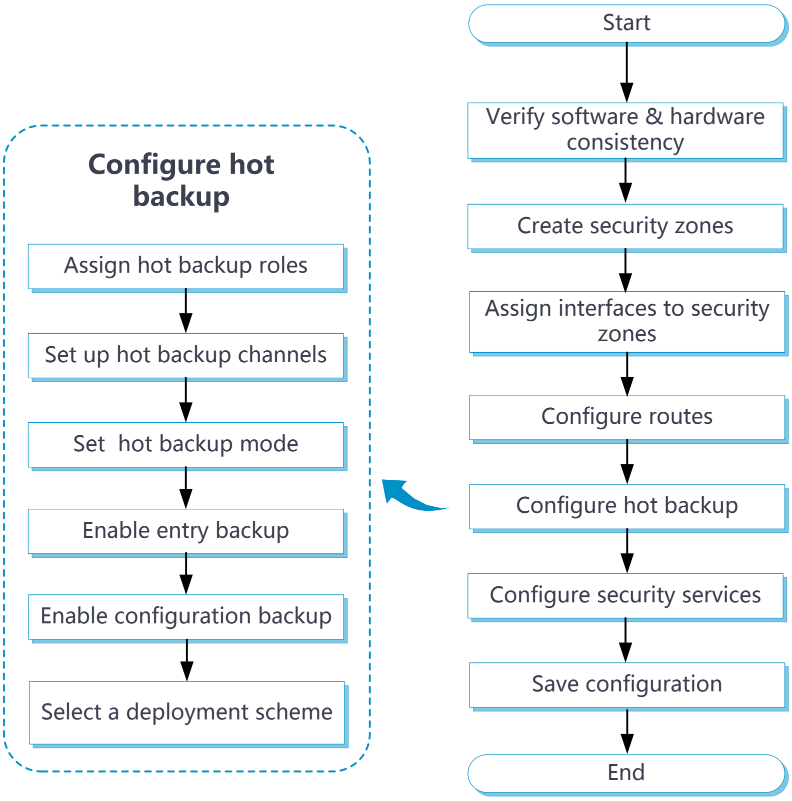
The device by default permits the hot backup packets transmitted over the hot backup channels. You do not need to configure security policies for the hot backup packets.

Configure a feature prior to hot backup deployment if hot backup cannot back up configuration of the feature. For more information about the features that can be backed up by hot backup, see "Configuration backup."

Hot backup configuration flow

Figure 11 shows the configuration flow for hot backup.

Figure 11 Hot backup configuration flow chart


Start
Verify software & hardware consistency
Create security zones
Assign interfaces to security zones
Configure routes
Configure hot backup
Configure security services
Save configuration
End
Configure hot backup
Assign hot backup roles
Set up hot backup channels
Set hot backup mode
Enable entry backup
Enable configuration backup
Select a deployment scheme

Hot backup tasks at a glance

To configure hot backup, perform the following tasks:

1. Configuring the hot backup role
2. Configuring hot backup data synchronization settings
    a. Configuring the hot backup control channel
    b. Configuring the hot backup data channel
    c. Enabling service entry hot backup
    d. Configuring hot backup configuration synchronization
3. Enabling traffic switchover upon failure recovery
4. Setting the hot backup mode

    Choose one of the following tasks:
    - Configuring the active/standby mode
    - Configuring the dual-active mode
5. Configuring hot backup associations

    Choose one of the following tasks:
    - Associating hot backup with VRRP
    - Associating hot backup with routing protocols
    - Configuring hot backup transparent in-path deployment
6. (Optional.) Associating hot backup with Track
7. (Optional.) Performing an active/standby switchover

8. (Optional.) Enabling transparent service traffic transmission between the hot backup members
9. (Optional.) Configuring service features on the hot backup system

Configuring the hot backup role

About this task

Hot backup backs up important configuration from the primary device to the secondary device to prevent service interruption when an active/standby switchover occurs. The configuration on the secondary device is overwritten. The unidirectional backup mechanism avoids configuration conflicts, especially in dual-active mode. The hot backup roles can only be manually assigned to devices.

Each hot backup member device adds a prefix to the view prompt to identify its hot backup role.

- The primary device adds the **RBM_P** prefix, **RBM_P<Sysname>** for example.
- The secondary device adds the **RBM_S** prefix, **RBM_S<Sysname>** for example.

After you assign hot backup roles to the hot backup member devices, both devices add the **RBM_P** prefix to their view prompts. The devices display view prompt prefixes according to their hot backup roles after they set up the hot backup control channel.

Restrictions and guidelines

You must assign the primary and secondary roles to the two member devices in the hot backup system, respectively.

As a best practice to ensure correct operation of the hot backup system, enable configuration backup on the primary device.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

   remote-backup group

3. Configure the hot backup role.

   device-role { **primary** | **secondary** }

   By default, the hot backup role is not configured.

Configuring the hot backup control channel

About this task

Hot backup compares the specified local and peer IP address to determine the device role for setting up the control channel. The device with higher IP address acts as the server, and the other device acts as the client to initiate the TCP connection.

If the port number is configured on the server, the port provides services for the client. If the port number is configured on the client, the port serves as the destination port to establish TCP connection to the server. The source port is randomly generated on the client.

Restrictions and guidelines

You can specify only one peer IP address with the same port number on the hot backup member devices. The specified port cannot be the same as the TCP listening port in use.

The local and peer IP addresses used for setting up the control channel cannot be identical.

You can set up an IPv4 control channel or IPv6 control channel, but not both.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

   remote-backup group

3. Set up a hot backup control channel. Choose one of the following options:
   - Set up an IPv4 control channel.
     - Configure the peer IPv4 address for setting up the hot backup control channel.

       remote-ip *ipv4-address* [**port** *port-number*]

       By default, the peer IPv4 address is not configured.
     - Configure the local IPv4 address for setting up the hot backup control channel.

       local-ip *ipv4-address*

       By default, the local IPv4 address is not configured.
   - Set up an IPv6 control channel.
     - Configure the peer IPv6 address for setting up the hot backup control channel.

       remote-ipv6 *ipv6-address* [**port** *port-number*]

       By default, the peer IPv6 address is not configured.
     - Configure the local IPv6 address for setting up the hot backup control channel.

       local-ipv6 *ipv6-address*

       By default, the local IPv6 address is not configured.
4. Set the interval for sending hot backup keepalive packets.

   keepalive interval *interval*

   By default, the device sends hot backup keepalive packets at one-second intervals.
5. Set the maximum number of hot backup keepalive attempts.

   keepalive count *counts*

   By default, the maximum number of hot backup keepalive attempts is 10.

Configuring the hot backup data channel

1. Enter system view.

   system-view
2. Enter RBM view.

   remote-backup group
3. Configure a hot backup data channel.

   data-channel interface *interface-type interface-number*

   By default, no hot backup data channel is configured.

Enabling service entry hot backup

About this task

Service entry hot backup enables the active device to back up service entries to the standby device in real time.

Enable HTTP and DNS backup if asymmetric-path traffic traverses the hot backup system. HTTP and DNS backup ensures that a flow and its return traffic are processed correctly on the hot backup members.

If hot backup active/standby mode is used or only symmetric-path traffic traverses the hot backup system, disabling HTTP and DNS backup can improve performance of the hot backup members at the expense of delayed data synchronization. When you disable HTTP and DNS backup, make sure you are fully aware of the impact on the network. A device removes a DNS or HTTP connection if packet exchange is inactive. When a switchover interrupts a connection, the DNS or HTTP client re-initiates the connection immediately, which has little impact on user services.

Restrictions and guidelines

When the hot backup system is operating stably and is processing traffic, do not execute the **`reset session table`** command. If you execute the command, traffic might be interrupted, and the session entries might become inconsistent on the primary and backup devices. For more information about session management, see *Security Configuration Guide*.

Procedure

1. Enter system view.
   `system-view`
2. Enter RBM view.
   `remote-backup group`
3. Enable service entry hot backup.
   `hot-backup enable`

   By default, service entry hot backup is enabled.
4. Enable hot backup for the session entries of application layer protocols.
   `hot-backup protocol { dns | http } * enable`

   By default, the hot backup system performs hot backup for the session entries of application layer protocols.

   The **`hot-backup protocol enable`** command takes effect only on DNS and HTTP. The device automatically backs up the session entries of other application layer protocols after you enable service entry hot backup.

Configuring hot backup configuration synchronization

About this task

The hot backup member devices can synchronize configuration changes in real time and synchronize all configuration in bulk.

- **Real-time synchronization**—The primary device copies added, deleted, or modified configuration to the secondary device in real time to maintain configuration consistency.
- **Bulk synchronization**—The primary device backs up important configuration in bulk to the secondary device. The secondary device will delete the settings that are inconsistent with those on the primary device.

Hot backup can perform hot backup configuration synchronization only when automatic configuration synchronization is enabled and the hot backup control channel is set up.

A bulk synchronization is triggered by the following events:

- The hot backup control channel is set up and automatic configuration synchronization is enabled (including the default status) for the first time. The primary device will send all its key configuration in bulk to overwrite the configuration on the secondary device. Hot backup does not perform bulk synchronization afterwards, even if automatic configuration synchronization is enabled again or the hot backup control channel is set up again.

- An hot backup member device reboots, or the hot backup process restarts, and automatic configuration synchronization is enabled on the other device. After the reboot or restart and the hot backup control channel is set up again, the device that does not reboot will send all its key configuration in bulk to overwrite the configuration on the rebooted device.

Restrictions and guidelines

After the hot backup control channel is set up and automatic configuration synchronization is enabled for the first time, do not disable automatic configuration synchronization. If you fail to do so, the devices will not perform bulk configuration, and configuration inconsistency will affect services.

If the amount of configuration to be synchronized is large, bulk synchronization might take one to two hours. To avoid the issue, you can perform one of the following operations:

- Enable automatic configuration synchronization first when you configure the hot backup system.
- Copy the configuration file to the secondary device during initial network deployment and then enable configuration consistency check.

Procedure

1. Enter system view.

   `system-view`

2. Enter RBM view.

   `remote-backup group`

3. Enable automatic configuration synchronization.

   `configuration auto-sync enable`

   By default, automatic configuration synchronization is enabled.

4. Enable configuration consistency check.

   `configuration sync-check` `[` **`interval`** *`interval`* `]`

   By default, configuration consistency check is enabled.

5. (Optional.) Perform a one-off configuration consistency check.

   `configuration manual-sync-check`

   This command applies only to the primary device.

6. (Optional.) Manually synchronize the configuration of the primary device to the secondary device.

   `configuration manual-sync`

   This command applies only to the primary device.

Enabling traffic switchover upon failure recovery

About this task

After an active/standby switchover occurs, if the original active device recovers, traffic will not be switched back by default. Perform this task to enable traffic switchover to the original active device upon failure recovery. You can set a delay timer to ensure smooth service switchover.

Restrictions and guidelines

In dual-active mode, you must enable this feature to ensure that both devices can operate after the failure is recovered.

This feature does not take effect on ongoing traffic switchovers. It applies only to subsequent traffic switchovers.

Procedure

1. Enter system view.
   system-view
2. Enter RBM view.
   remote-backup group
3. Enable traffic switchover upon failure recovery.
   delay-time *delay-time*
   By default, traffic switchover upon failure recovery is disabled.

Configuring the active/standby mode

1. Enter system view.
   system-view
2. Enter RBM view.
   remote-backup group
3. Configure the active/standby mode.
   undo backup-mode dual-active
   By default, the hot backup mode is active/standby.

Configuring the dual-active mode

1. Enter system view.
   system-view
2. Enter RBM view.
   remote-backup group
3. Configure the dual-active mode.
   backup-mode dual-active
   By default, the hot backup mode is active/standby.

Associating hot backup with VRRP

Restrictions and guidelines

You can associate only VRRP groups operating in standard mode with hot backup.

Use hot backup in combination with VRRP groups only for in-path hot backup deployment between upstream and downstream switches. As a best practice, assign the primary device to the VRRP active group, and assign the secondary device to the VRRP standby group.

You can assign multiple virtual IP addresses to a VRRP group associated with hot backup. Make sure the interfaces in the VRRP group do not own the virtual IP addresses.

You cannot associate a VRRP group with both hot backup and Track.

You cannot use the **track vlan** command in conjunction with the **track interface** command.

For an IPv6 VRRP group to work correctly, you must assign it a link-local address and a global unicast address as virtual IPv6 addresses.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Associate hot backup with VRRP. Choose one of the following options:
   - Create an IPv4 VRRP group and associate it with hot backup.

     vrrp vrid *virtual-router-id* **virtual-ip** *virtual-address* [*mask* | *mask-length*] { **active** | **standby** }

     By default, no IPv4 VRRP groups exist.
   - Create an IPv6 VRRP group and associate it with hot backup. Assign a link-local address and a global unicast address as virtual IPv6 addresses to the VRRP group.

     vrrp ipv6 vrid *virtual-router-id* **virtual-ip** *virtual-address* [*prefix-length*] **link-local** { **active** | **standby** }

     vrrp ipv6 vrid *virtual-router-id* **virtual-ip** *virtual-address*

     By default, no IPv6 VRRP groups exist.

Associating hot backup with routing protocols

Enabling hot backup to adjust the link cost for a routing protocol

About this task

When you use hot backup together with a routing protocol, you can enable hot backup to adjust the link cost for a routing protocol on the standby device. This feature applies to the scenario where the hot backup member devices run the same routing protocol. If you configure this feature, the active device advertises the original link costs for a routing protocol, and the standby device advertises one of the following link costs:

- **Absolute cost**—The device advertises an absolute link cost for the routing protocol.
- **Calculated cost**—The device advertises the original link cost plus the configured increment cost for the specified routing protocol.

The feature takes effect on only the standby device.

Restrictions and guidelines

This feature applies to the scenario where the hot backup member devices run the same routing protocol.

To ensure switchover of both uplink and downlink traffic to the new active device, configure this feature with the same parameters on both hot backup member devices.

Do not use the **silent-backup-interface** and **adjust-cost enable** commands together.

On a hot backup system in dual-active mode and in collaboration with a routing protocol, configure per-flow load sharing for IP forwarding on both the uplink and downlink devices as a best practice.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

```
remote-backup group
```

3. Enable hot backup to adjust the link cost for the specified routing protocol on the standby device.

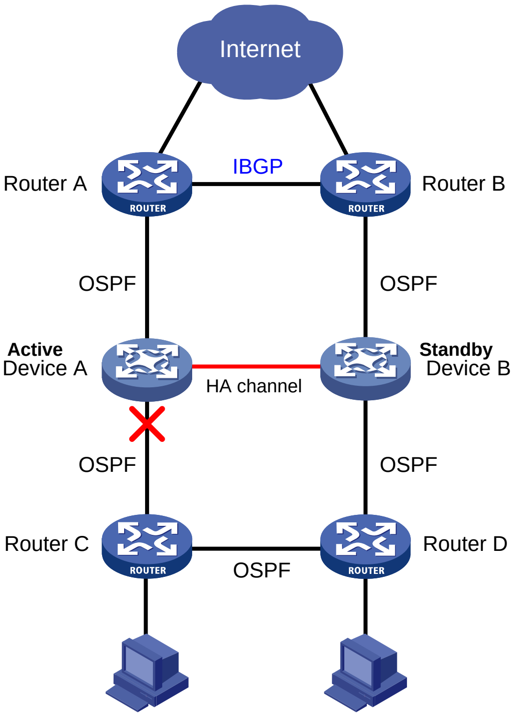
   adjust-cost { **bgp** | **isis** | **ospf** | **ospfv3** } **enable** { **absolute** [*absolute-cost*] | **increment** [*increment-cost*] }

   By default, hot backup does not adjust the link cost for the specified routing protocol on the standby device.

Disabling the standby device from sending or receiving protocol packets of a routing protocol.

This feature applies to the scenario where the hot backup member devices run multiple routing protocols. As shown in Figure 12, IBGP and OSPF have different default priorities. When the link between Device A and Router C fails, traffic destined for the hosts might be forwarded to Device A as OSPF has higher priority than IBGP. To resolve this issue, disable Device A from sending or receiving OSPF protocol packets to disconnect its OSPF neighbors. Then, Device B will forward all uplink and downlink traffic.

Figure 12 hot backup member devices running multiple routing protocols



Restrictions and guidelines

This feature applies to the scenario where the hot backup member devices run multiple routing protocols.

To ensure switchover of both uplink and downlink traffic to the new active device, configure this feature with the same parameters on both hot backup member devices.

Do not use the **silent-backup-interface** and **adjust-cost enable** commands together.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RBM view.

   remote-backup group

3. Disable the standby device from sending or receiving protocol packets of a dynamic routing protocol.

   silent-backup-interface { **ospf** | **ospfv3** }

   By default, the standby device can send and receive protocol packets of a dynamic routing protocol.

Configuring hot backup transparent in-path deployment

Enabling hot backup to monitor interfaces

About this task

Perform this task to enable hot backup to monitor the interfaces connecting the uplink and downlink devices in hot backup transparent in-path deployment. The monitored interfaces can forward packets only when they are all up. If any of the monitored interfaces goes down, none of them will be able to forward packets.

Restrictions and guidelines

You can use the **track interface** and **track** commands in conjunction, but you cannot use these commands to monitor the same interfaces.

The **track vlan** and **track interface** commands are mutually exclusive. You cannot configure both of them.

Hot backup does not support monitoring member ports of aggregate interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

   remote-backup group

3. Enable hot backup to monitor a Layer 2 or Layer 3 Ethernet interface.

   track interface *interface-type interface-number*

   By default, hot backup does not monitor any interfaces.

Enabling hot backup to monitor VLANs

About this task

Perform this task to enable hot backup to monitor the VLANs of the uplink and downlink devices in hot backup transparent in-path deployment. The monitored VLANs are active and the member ports can forward packets only when the member ports are all up. If any of the member ports goes down, none of them will be able to forward packets, and all the monitored VLANs will become inactive.

In active/standby mode, the state of monitored VLANs is active on the primary device and inactive on the secondary device.

In dual-active mode, the state of monitored VLANs is active on both the primary and secondary devices.

Restrictions and guidelines

The **track vlan** command is mutually exclusive with the **track interface** and **track** commands. You cannot use the **track vlan** command in conjunction with the **track interface** or **track** command.

Do not enable hot backup to monitor VLAN 1 (to which all access ports belong by default). This restriction prevents an unused interface in down state from interrupting operation of other interfaces in VLAN 1.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

   remote-backup group

3. Enable hot backup to monitor a VLAN.

   track vlan *vlan-id*

   By default, hot backup does not monitor any VLANs.

Associating hot backup with Track

About this task

Perform this task to associate hot backup with Track to monitor links. If one of the monitored track entries becomes Negative, hot backup performs an active/standby switchover and switches traffic to the new active device to ensure service continuity. For more information about Track, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter RBM view.

   remote-backup group

3. Associate hot backup with Track.

   track *track-entry-number*

   By default, hot backup is not associated with Track.

Performing an active/standby switchover

About this task

If you want to replace components or upgrade software on the current active device, you can perform this task to switch services to the standby device.

Restrictions and guidelines

This feature applies only when hot backup operates in active/standby mode, and it takes effect on only the active device.

In hot backup and VRRP associated network, performing this task might cause temporary virtual IP address conflict in the VRRP group, which is considered a normal condition.

For stable operation of hot backup, do not repeatedly perform this task within one minute.

Procedure

1. Enter system view.

   `system-view`

2. Enter RBM view.

   `remote-backup group`

3. Perform an active/standby switchover.

   `switchover request`

Enabling transparent service traffic transmission between the hot backup members

About this task

Enable transparent service traffic transmission only when asymmetric-path traffic traverses the hot backup system operating in dual-active mode.

If an asymmetric-path flow traverses the hot backup system operating in dual-active mode, the flow and its return traffic are processed by different hot backup members. This will degrade the traffic processing performance of modules such as NBAR, DPI, and load balancing. For example, the packet recognition rate of NBAR might drop. For an asymmetric-path flow and its return traffic to be processed by the same hot backup member, enable transparent service traffic transmission. Transparent service traffic transmission is resource-intensive. Make sure you are fully aware of the impact of this feature when you use it on a live network.

Procedure

1. Enter system view.

   `system-view`

2. Enter RBM view.

   `remote-backup group`

3. Enable transparent service traffic transmission between the hot backup members.

   `transparent-transmit enable`

   By default, transparent service traffic transmission is enabled*.*

Configuring service features on the hot backup system

NAT on the hot backup system

For NAT to operate correctly on the hot backup system, you must associate NAT features with the VRRP groups. For example, when you use dynamic NAT, static NAT, NAT server, or NAT444, you must associate the feature with the VRRP groups. For more information about NAT features, see *NAT Configuration Guide*.

NAT features have similar mechanisms, and the operating mode of hot backup does not change the IP address translation process. This section uses dynamic NAT on the hot backup system in active/standby mode to explain how NAT works on the hot backup system.

About NAT on the hot backup system

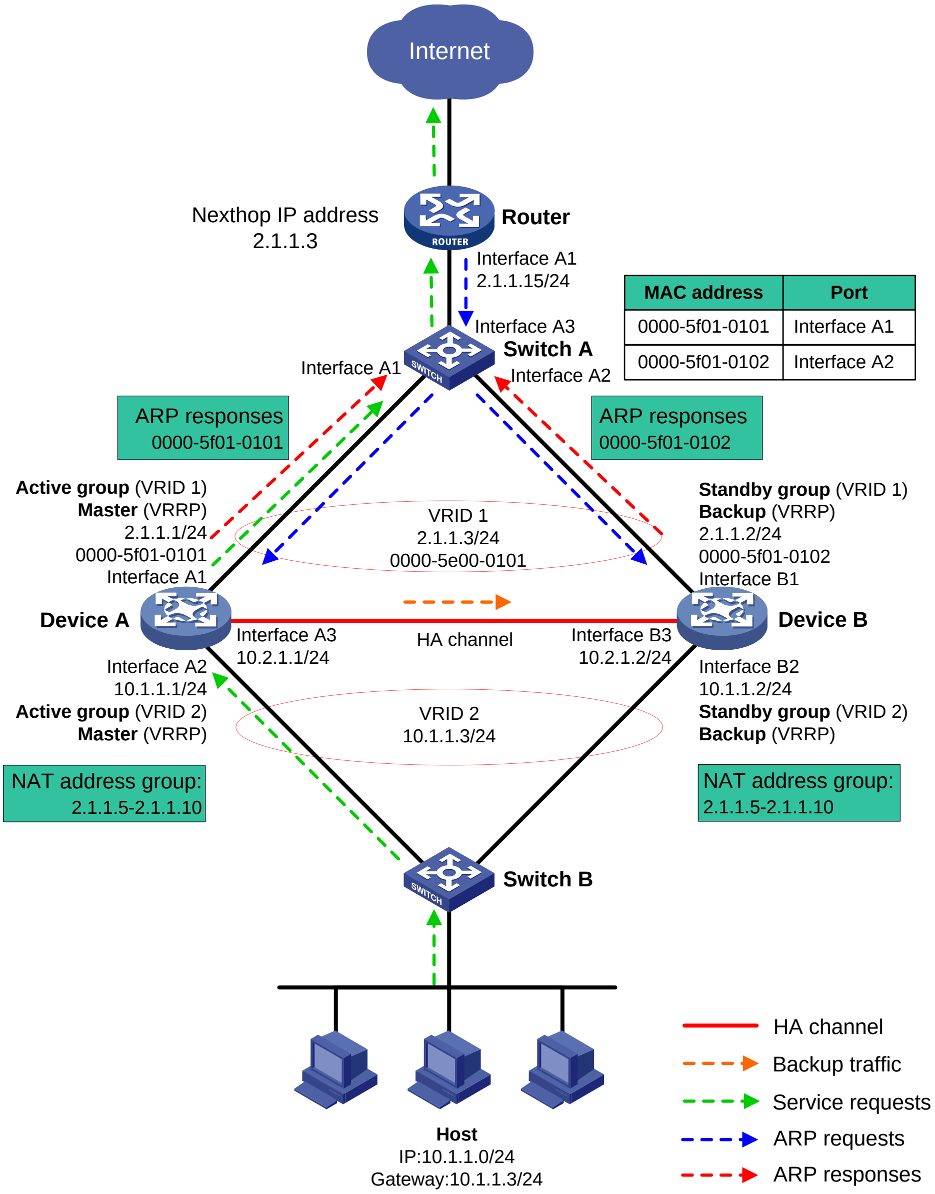
When receiving an ARP request with a target IP address that belongs to the subnet of the IP address of a NAT interface, a NAT device replies with the MAC address of the NAT interface.

As shown in Figure 13, dynamic NAT is configured on the hot backup system that is operating in active/standby mode. If dynamic NAT is not associated with any VRRP group, the devices process the traffic as follows when the host accesses the Internet:

1. When receiving the packets sent by the host, Device A translates the source IP address into a public IP address in the NAT address group and forwards the packets to the router. In this example, the public IP address is in the same subnet as the virtual IP address of uplink VRRP group 1.
2. The router receives the return packets and broadcasts an ARP request for the destination public IP address.
3. Device A and Device B receive the ARP request and reply with the MAC address of their respective uplink interface because they have the same NAT address group configuration.
4. The router might send the return packets to the uplink interface of Device A or Device B, which affects service continuity.

For the router to learn the virtual MAC address of the uplink VRRP group, you must associate NAT features with the VRRP group.

Figure 13 NAT not associated with a VRRP group



MAC address	Port
0000-5f01-0101	Interface A1
0000-5f01-0102	Interface A2

Traffic forwarding process

The master in a VRRP group relies with the virtual MAC address of the VRRP group to an ARP request if the following requirements are met:

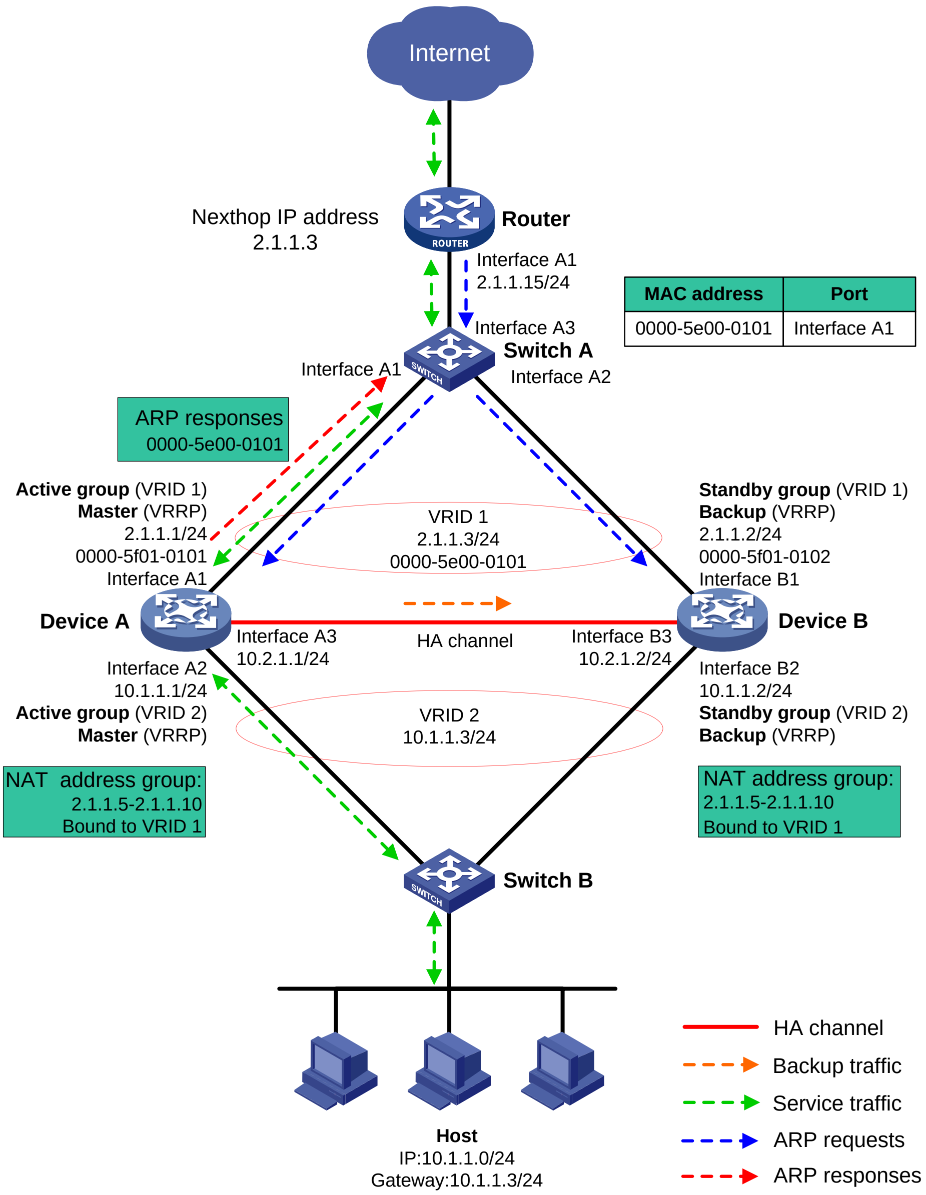
- NAT features are associated with the VRRP group.
- The target IP address belongs to the subnet that contains the IP address of a NAT interface.

As shown in Figure 14, dynamic NAT is configured on the hot backup system that is operating in active/standby mode. If NAT is associated with the uplink VRRP group, the devices process the traffic as follows when the host accesses the Internet:

1. When receiving the packets sent by the host, Device A translates the source IP address into a public IP address in the NAT address group and forwards the packets to the router. In this example, the public IP address is in the same subnet as the virtual IP address of uplink VRRP group 1.
2. The router receives the return packets and broadcasts an ARP request for the destination public IP address.

3. Device A and Device B receive the ARP request, and Device A (master) replies with the virtual MAC addresses of the uplink VRRP group.
4. Router A sends the return packets to Device A.

Figure 14 NAT on the hot backup system



MAC address	Port
0000-5e00-0101	Interface A1

For more information about VRRP group association with dynamic NAT, static NAT, NAT server, and NAT444, see *Layer 3—IP Services Configuration Guide*.

Hot backup support for SSL VPN

The hot backup channels are used to back up SSL VPN data, including user data, entries, and configuration. For more information about SSL VPN configuration, see *VPN Configuration Guide*.

Hot backup supports SSL VPN only when it is operating in active/standby mode and collaborating with VRRP groups.

Hot backup support for DPI services

When you use DPI services on the hot backup system operating in dual-active mode, you must enable DPI support for hot backup if asymmetric-path traffic exists. If you do not enable this feature, DPI services might fail to correctly identify and process packets. For more information about DPI support for hot backup, see DPI engine configuration in *DPI Configuration Guide*.

Hot backup support for contexts

You can configure hot backup only on the default context. The configuration will be issued to all non-default contexts.

To ensure configuration and service entry consistency, all non-default contexts use the hot backup channels created for the default context to back up configuration and service entries and transmit service traffic.

All non-default contexts use the keepalive detection and configuration consistency check mechanisms of the default context. When an active/standby switchover occurs on any context, the other contexts will perform an active/standby switchover accordingly.

Hot backup deployment schemes

Hot backup supports the following deployment schemes:

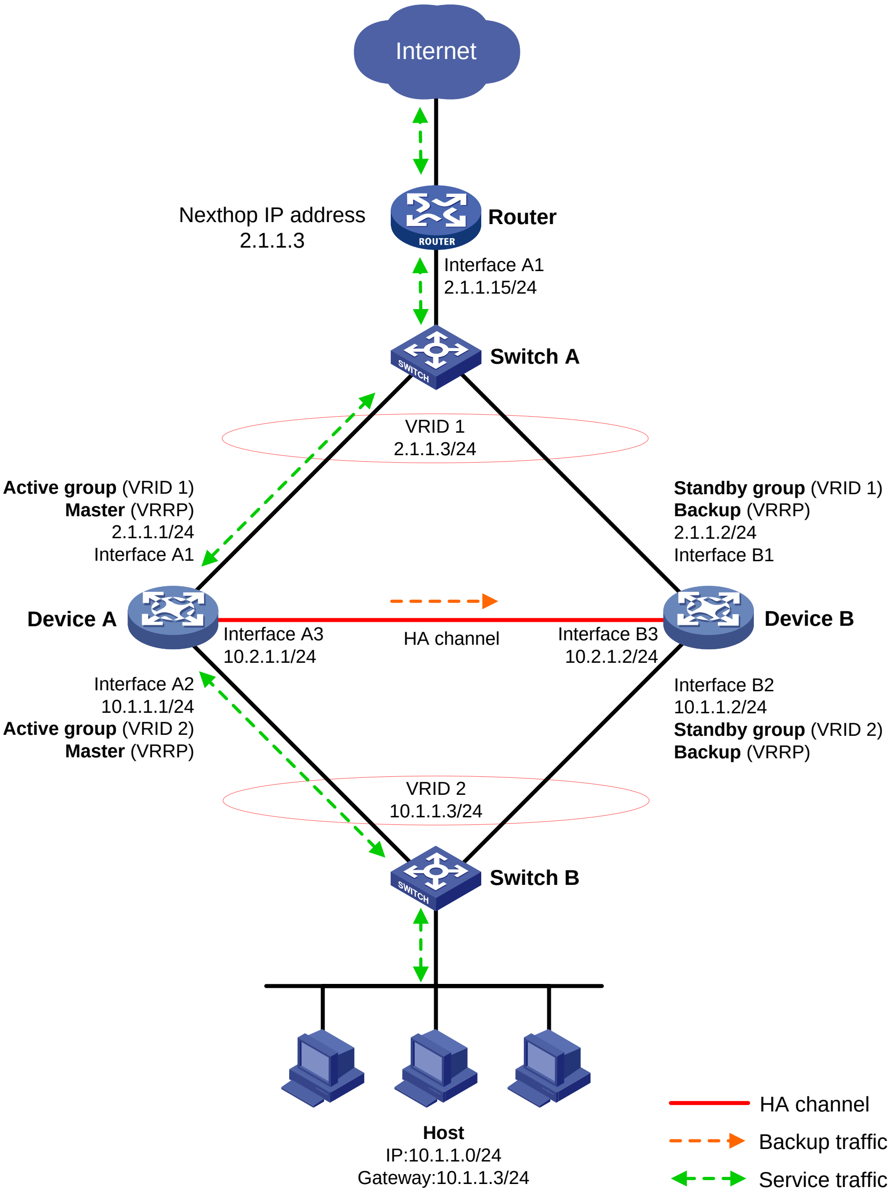
- Routed in-path deployment in active/standby mode.
- Routed in-path deployment in dual-active mode.
- Transparent in-path deployment in active/standby mode.
- Transparent in-path deployment in dual-active mode.

Routed in-path deployment in active/standby mode

Figure 15 shows a typical model of routed in-path deployment in active/standby mode. A hot backup system is directly connected to the upstream and downstream Layer 2 switches by Layer 3 interfaces. To use this scheme in collaboration with VRRP, perform the following tasks:

- Establish hot backup channels between Device A and Device B.
- On Device A and Device B, create uplink VRRP group 1 and downlink VRRP group 2 and associate them with hot backup.
- On Device A, associate VRRP group 1 and VRRP group 2 with the VRRP active group. On Device B, associate VRRP group 1 and VRRP group 2 with the VRRP standby group.
- On Device A and Device B, specify the IP address of Interface A1 on the router (2.1.1.15) as the next hop of the route to the Internet.
- On the router, specify the virtual IP address of VRRP group 1 (2.1.1.3) as the next hop of the route to the host's subnet.
- On the host, specify the virtual IP address of VRRP group 2 (10.1.1.3) as the default gateway.
- On Switch A, assign the interfaces attached to the router, Device A, and Device B to the same VLAN.
- On Switch B, assign the interfaces attached to the host, Device A, and Device B to the same VLAN.

Figure 15 Routed in-path deployment in active/standby mode



The following shows how traffic is forwarded when the host accesses the Internet in Figure 15:

1. The host identifies that the destination IP address is on a different subnet and sends an ARP request to obtain the MAC address of the default gateway. In this example, the host does not have the ARP entry for the default gateway.
2. Switch B broadcasts the ARP request and learns the MAC address of the host.
3. Device A and Device B receives the ARP request, and Device A (master) replies with the virtual MAC address of VRRP group 2.
4. Switch B learns the MAC address entry for the virtual MAC address of VRRP group 2 and forwards the ARP reply to the host.
5. The host learns the virtual MAC address and sends the packets destined for the Internet to the default gateway.
6. Switch B forwards the packets to Device A (master). The traffic of the host will be processed and forwarded by Device A as long as it is the master.

7. Device A does not have the ARP entry for the next hop of the route to the Internet and sends an ARP request to obtain the MAC address of the next hop. In the ARP request, the source MAC address is the virtual MAC address of VRRP group 1.
8. Switch A and the router then perform typical forwarding and ARP and MAC address learning.

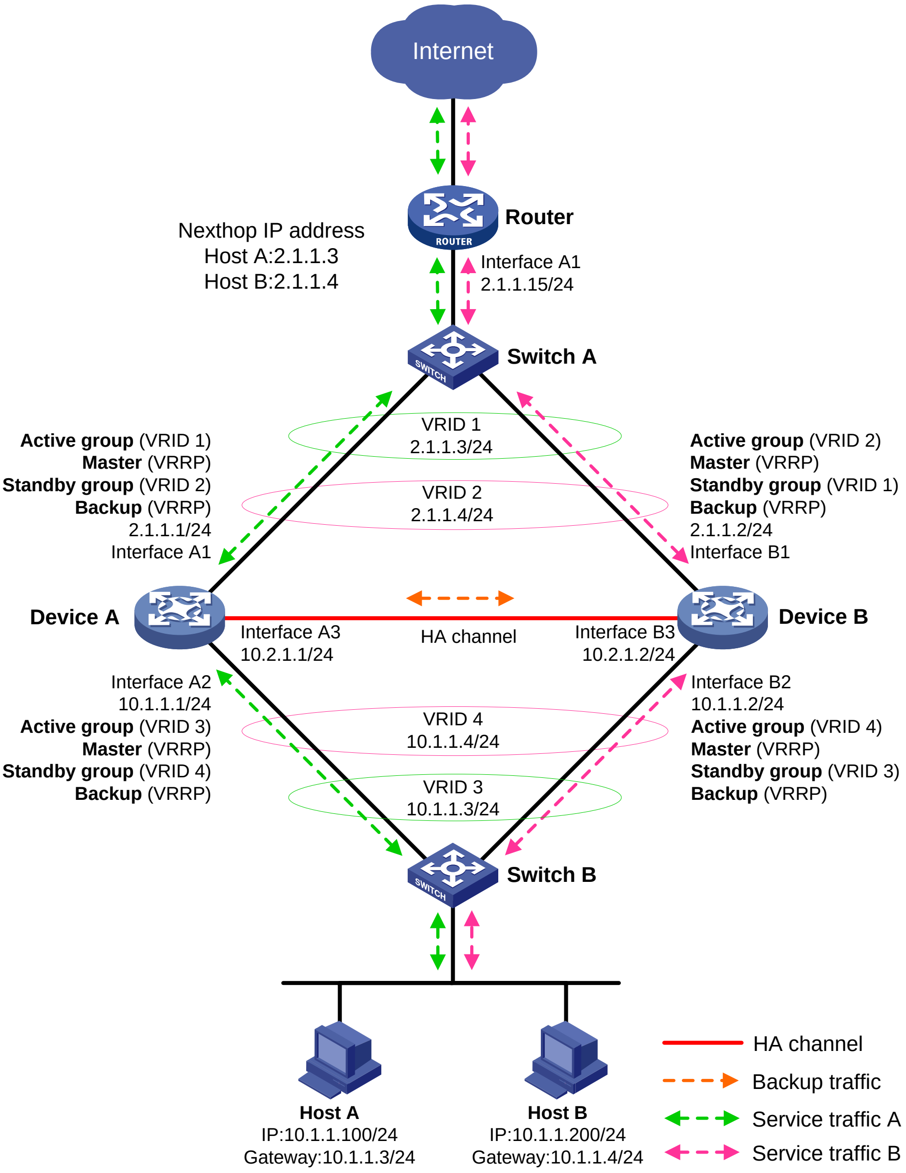
The forwarding process for the traffic sent from the Internet to the host is similar to the above process.

Routed in-path deployment in dual-active mode

Figure 16 shows a typical model of routed in-path deployment in dual-active mode. A hot backup system is directly connected to the upstream and downstream Layer 2 switches by Layer 3 interfaces. To use this scheme in collaboration with VRRP, perform the following tasks:

- Establish hot backup channels between Device A and Device B.
- On Device A and Device B, create two uplink VRRP groups and two downlink VRRP groups.
- Create VRRP group 3 and VRRP group 4 on the downlink interfaces of Device A and Device B.
- On Device A, associate VRRP group 1 and VRRP group 3 with the VRRP active group, and associate VRRP group 2 and VRRP group 4 with the VRRP standby group.
- On Device B, associate VRRP group 1 and VRRP group 3 with the VRRP standby group, and associate VRRP group 2 and VRRP group 4 with the VRRP active group.
- On Device A and Device B, specify the IP address of Interface A1 on the router (2.1.1.15) as the next hop of the route to the Internet.
- On the router, configure routes as follows:
  - Specify the virtual IP address of VRRP group 1 (2.1.1.3) as the next hop of the route to Host A's subnet.
  - Specify the virtual IP address of VRRP group 2 (2.1.1.4) as the next hop of the route to Host B's subnet.
- On Host A, specify the virtual IP address of VRRP group 3 (10.1.1.3) as the default gateway.
- On Host B, specify the virtual IP address of VRRP group 4 (10.1.1.4) as the default gateway.
- On Switch A, assign the interfaces attached to the router, Device A, and Device B to the same VLAN.
- On Switch B, assign the interfaces attached to the hosts, Device A, and Device B to the same VLAN.

Figure 16 Routed in-path deployment in dual-active mode



As shown in Figure 16, the traffic of Host A and Host B is distributed to Device A and Device B, respectively. The traffic forwarding process is similar to that in active/standby mode.

Transparent in-path deployment in active/standby mode

Figure 17 shows a typical model of transparent in-path deployment in active/standby mode. The Layer 2 hot backup system is directly connected to the upstream and downstream Layer 2 switches by Layer 2 interfaces. To use this scheme, perform the following tasks:

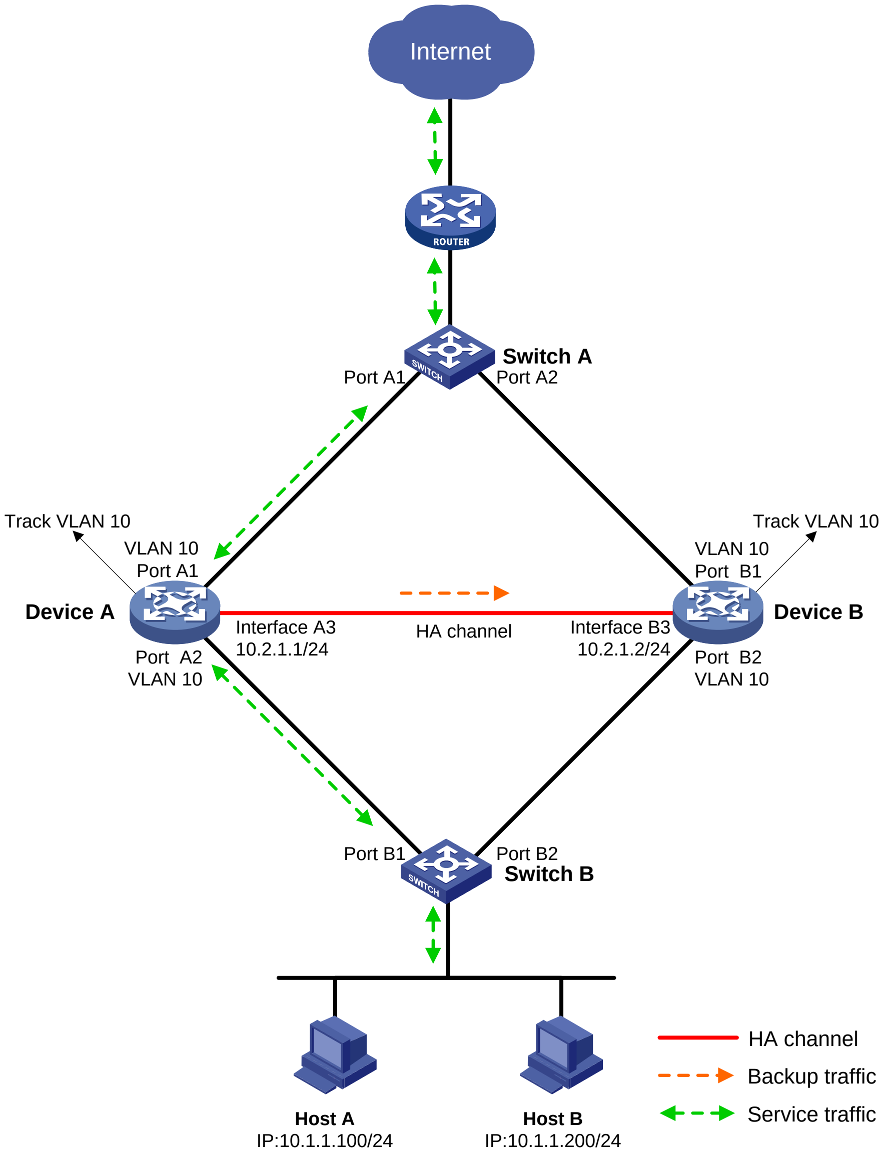
- Establish hot backup channels between Device A and Device B.
- On Device A and Device B, assign uplink and downlink interfaces to the same VLAN.
- Configure hot backup to monitor one of the following objects:
  - The VLAN where Device A and Device B reside.

    You do not need to enable spanning tree on the upstream and downstream switches.
  - Uplink and downlink interfaces of Device A and Device B.

You must enable spanning tree on the upstream and downstream switches.

- On Switch A, assign the interfaces attached to the upstream router, Device A, and Device B to the same VLAN.
- On Switch B, assign the interfaces attached to the hosts, Device A, and Device B to the same VLAN.

Figure 17 Transparent in-path deployment in active/standby mode



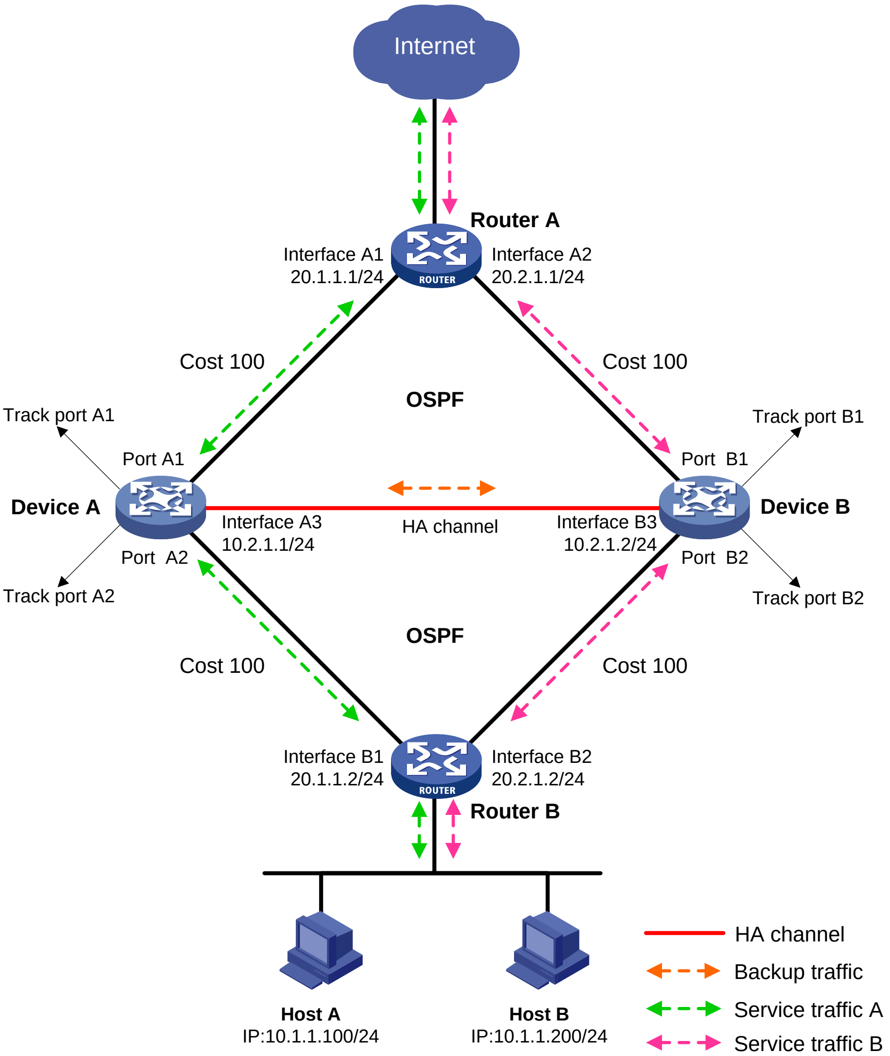
Transparent in-path deployment in dual-active mode

Figure 18 shows a typical model of transparent in-path deployment in dual-active mode. The Layer 2 hot backup system is directly connected to the upstream and downstream routers by Layer 2 interfaces. To use this scheme, perform the following tasks:

- Establish hot backup channels between Device A and Device B.
- On Device A and Device B, assign uplink and downlink interfaces to the same VLAN.
- On Device A and Device B, configure hot backup to monitor the status of the uplink and downlink interfaces.

- On Router A and Router B, configure the same cost for OSPF routes and enable per-flow load sharing among ECMP routes.

Figure 18 Transparent in-path deployment in dual-active mode



Display and maintenance commands for hot backup

Execute **display** commands in any view.

Task	Command
Display hot backup status information.	**display remote-backup-group status**
Display the configuration consistency check result for hot backup.	**display remote-backup-group sync-check**

Hot backup configuration examples (IPv4)

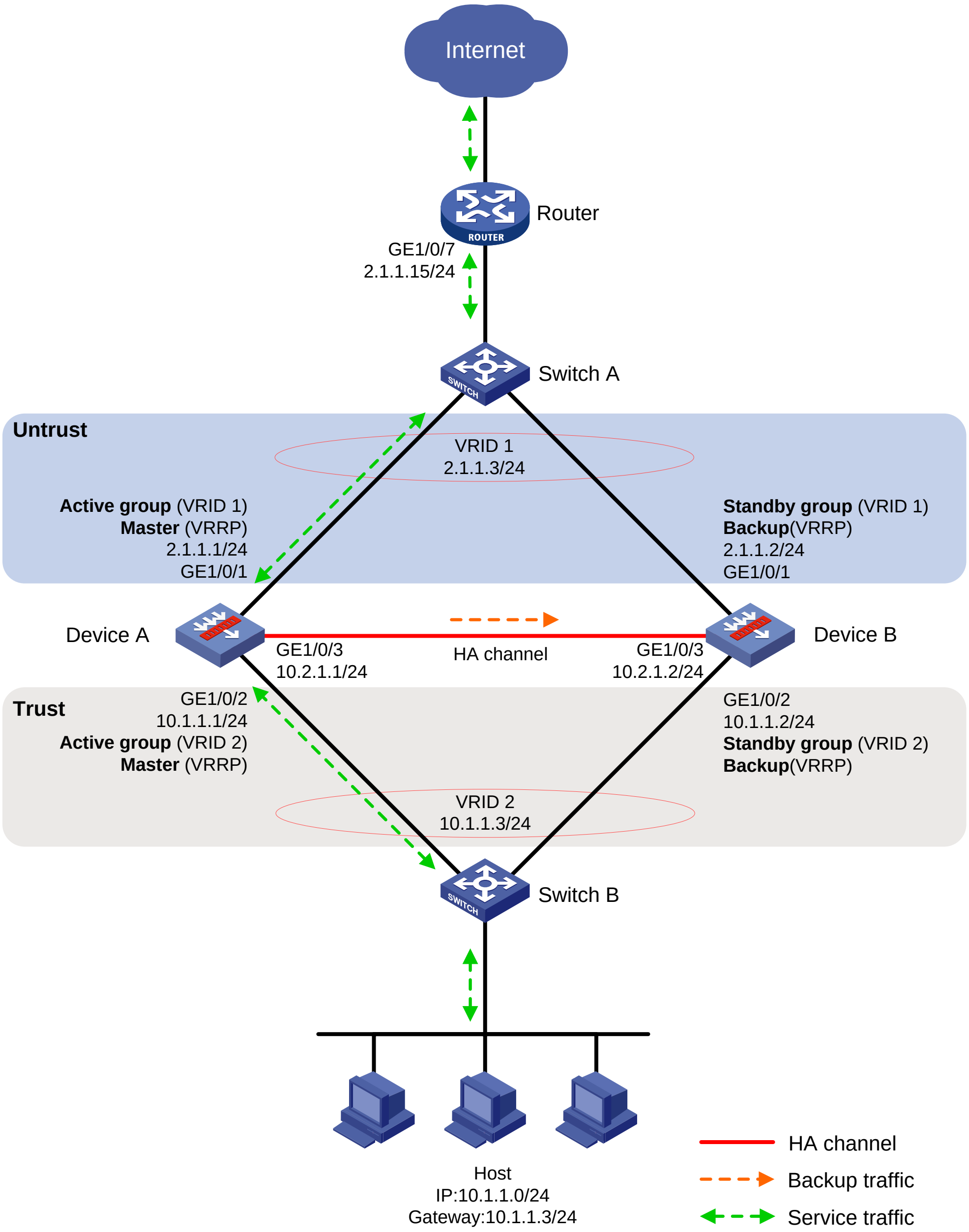
Example: Configuring active/standby hot backup in collaboration with VRRP

Network configuration

As shown in Figure 19, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.
- Configure the hot backup system to operate in active/standby mode.
- Configure Device A and Device B as the primary device and the secondary device, respectively.

Figure 19 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Switch A:

   NOTE:

   This step only provides the brief configuration procedure.

   # Create VLAN 10.

   # Configure the interfaces attached to the hot backup system and the router to operate at Layer 2. Assign them to VLAN 10 as access interfaces.
3. Configure Switch B:

NOTE:

This step only provides the brief configuration procedure.

# Create VLAN 10.

# Configure the interfaces attached to the hot backup system and the host to operate at Layer 2. Assign them to VLAN 10 as access interfaces.

4. Configure the router:

NOTE:

This step only provides the brief configuration procedure.

# Assign 2.1.1.15/24 to GigabitEthernet 1/0/7.

# Configure routes as follows:

- Specify 2.1.1.3 (virtual IP address of VRRP group 1) as the next hop of the routes to the internal network.
- Specify the IP address of the peer interface attached to the traffic outgoing interface as the next hop of the route to the Internet.

5. Configure Device A:

a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 2.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

c. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.1.1.15.

```
[DeviceA] ip route-static 0.0.0.0 0.0.0.0 2.1.1.15
```

d. Configure a security policy.

Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

# Configure a rule named **trust-untrust** to permit the packets from 10.1.1.0/24 to the Internet.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure rules to permit VRRP protocol packets. When the hot backup channel is disconnected, Device A and Device B can exchange VRRP protocol packets to elect a VRRP master.

```
[DeviceA-security-policy-ip] rule name vrrp1
[DeviceA-security-policy-ip-4-vrrp1] source-zone trust
[DeviceA-security-policy-ip-4-vrrp1] destination-zone local
[DeviceA-security-policy-ip-4-vrrp1] service vrrp
[DeviceA-security-policy-ip-4-vrrp1] action pass
[DeviceA-security-policy-ip-4-vrrp1] quit
[DeviceA-security-policy-ip] rule name vrrp2
[DeviceA-security-policy-ip-5-vrrp2] source-zone local
[DeviceA-security-policy-ip-5-vrrp2] destination-zone trust
[DeviceA-security-policy-ip-5-vrrp2] service vrrp
[DeviceA-security-policy-ip-5-vrrp2] action pass
[DeviceA-security-policy-ip-5-vrrp2] quit
[DeviceA-security-policy-ip] rule name vrrp3
[DeviceA-security-policy-ip-6-vrrp3] source-zone untrust
[DeviceA-security-policy-ip-6-vrrp3] destination-zone local
[DeviceA-security-policy-ip-6-vrrp3] service vrrp
[DeviceA-security-policy-ip-6-vrrp3] action pass
[DeviceA-security-policy-ip-6-vrrp3] quit
[DeviceA-security-policy-ip] rule name vrrp4
[DeviceA-security-policy-ip-7-vrrp4] source-zone local
[DeviceA-security-policy-ip-7-vrrp4] destination-zone untrust
[DeviceA-security-policy-ip-7-vrrp4] service vrrp
[DeviceA-security-policy-ip-7-vrrp4] action pass
[DeviceA-security-policy-ip-7-vrrp4] quit
[DeviceA-security-policy-ip] quit
```

e. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ip 10.2.1.2
[DeviceA-remote-backup-group] local-ip 10.2.1.1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] undo backup-mode
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] quit
```

# Create VRRP groups and associate them with the hot backup system.

```
RBM_P[DeviceA] interface gigabitethernet 1/0/1
RBM_P[DeviceA-GigabitEthernet1/0/1] vrrp vrid 1 virtual-ip 2.1.1.3 active
RBM_P[DeviceA-GigabitEthernet1/0/1] quit
RBM_P[DeviceA] interface gigabitethernet 1/0/2
RBM_P[DeviceA-GigabitEthernet1/0/2] vrrp vrid 2 virtual-ip 10.1.1.3 active
RBM_P[DeviceA-GigabitEthernet1/0/2] quit
```

f. Configure security services on Device A. (Details not shown.)

6. Configure Device B:

   a. Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 2.1.1.2 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   ```

   c. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.1.1.15.

   ```
   [DeviceB] ip route-static 0.0.0.0 0.0.0.0 2.1.1.15
   ```

   d. Configure hot backup settings.

   # Set up a hot backup system.

   ```
   [DeviceB] remote-backup group
   [DeviceB-remote-backup-group] remote-ip 10.2.1.1
   [DeviceB-remote-backup-group] local-ip 10.2.1.2
   [DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
   [DeviceB-remote-backup-group] device-role secondary
   RBM_S[DeviceB-remote-backup-group] undo backup-mode
   RBM_S[DeviceB-remote-backup-group] hot-backup enable
   RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
   RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
   RBM_S[DeviceB-remote-backup-group] quit
   ```

   # Create VRRP groups and associate them with the hot backup system.

   ```
   RBM_S[DeviceB] interface gigabitethernet 1/0/1
   RBM_S[DeviceB-GigabitEthernet1/0/1] vrrp vrid 1 virtual-ip 2.1.1.3 standby
   RBM_S[DeviceB-GigabitEthernet1/0/1] quit
   RBM_S[DeviceB] interface gigabitethernet 1/0/2
   RBM_S[DeviceB-GigabitEthernet1/0/2] vrrp vrid 2 virtual-ip 10.1.1.3 standby
   RBM_S[DeviceB-GigabitEthernet1/0/2] quit
   ```

7. On the host, specify 10.1.1.3 (virtual IP address of VRRP group 2) as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

   # Verify that the hot backup channels have been set up.

   ```
   RBM_P[DeviceA] display remote-backup-group status
   Remote backup group information:
     Backup mode: Active/standby
     Device management role: Primary
     Device running status: Active
     Data channel interface: GigabitEthernet1/0/3
   ```

```
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that Device A is the master in all VRRP groups.

```
RBM_P[DeviceA] display vrrp
IPv4 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   VRRP active group status : Master
   VRRP standby group status: Master
 Total number of virtual routers : 2
 Interface          VRID  State        Running Adver   Auth     Virtual
                                       Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Master       100     100     None     2.1.1.3
 GE1/0/2            2     Master       100     100     None     10.1.1.3
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device A generates log messages when the host communicates with the Internet. (Details not shown.)

2. Verify the configuration on Device B:

# Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Secondary
  Device running status: Standby
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                         Status change        Cause
    2021-06-22 13:33:33          Initial to Active    Interface status changed
```

# Verify that Device A is the backup in all VRRP groups.

```
RBM_S[DeviceB] display vrrp
IPv4 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   VRRP active group status : Backup
   VRRP standby group status: Backup
 Total number of virtual routers : 2
 Interface          VRID  State        Running Adver   Auth     Virtual
                                       Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Backup       100     100     None     2.1.1.3
 GE1/0/2            2     Backup       100     100     None     10.1.1.3
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device B does not generate log messages when the host communicates with the Internet. (Details not shown.)

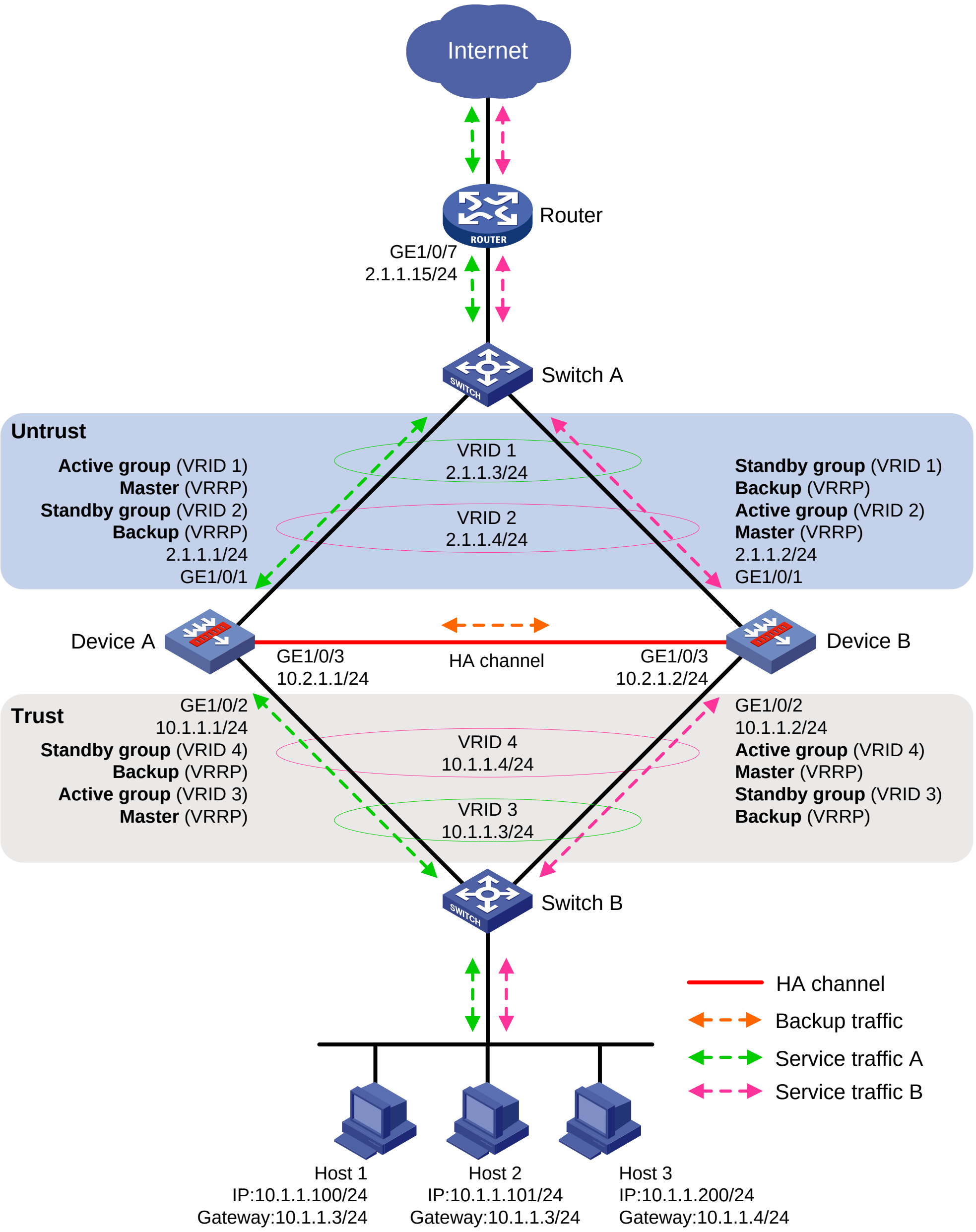
Example: Configuring dual-active hot backup in collaboration with VRRP

Network configuration

As shown in Figure 20, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.
- Configure the hot backup system to operate in dual-active mode.
- Configure Device A and Device B to load share traffic.

Figure 20 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Switch A:

   NOTE:

   This step only provides the brief configuration procedure.

   # Create VLAN 10.

   # Configure the interfaces attached to the hot backup system and the router to operate at Layer 2. Assign them to VLAN 10 as access interfaces.
3. Configure Switch B:

NOTE:

This step only provides the brief configuration procedure.

# Create VLAN 10.

# Configure the interfaces attached to the hot backup system and the host to operate at Layer 2. Assign them to VLAN 10 as access interfaces.

4. Configure the router:

NOTE:

This step only provides the brief configuration procedure.

# Assign 2.1.1.15/24 to GigabitEthernet 1/0/7.

# Configure routes as follows:

- Specify 2.1.1.3 (virtual IP address of VRRP group 1) as the next hop of the routes to some subnets of the internal network. Specify 2.1.1.4 (virtual IP address of VRRP group 2) as the next hop of the routes to the other subnets of the internal network.
- Specify the IP address of the peer interface attached to the traffic outgoing interface as the next hop of the route to the Internet.

5. Configure Device A:

a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 2.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

c. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.1.1.15.

```
[DeviceA] ip route-static 0.0.0.0 0.0.0.0 2.1.1.15
```

d. Configure a security policy.

Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

# Configure a rule named **trust-untrust** to permit the packets from 10.1.1.0/24 to the Internet.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

# Configure rules to permit VRRP protocol packets. When the hot backup channel is disconnected, Device A and Device B can exchange VRRP protocol packets to elect a VRRP master.

```
[DeviceA-security-policy-ip] rule name vrrp1
[DeviceA-security-policy-ip-4-vrrp1] source-zone trust
[DeviceA-security-policy-ip-4-vrrp1] destination-zone local
[DeviceA-security-policy-ip-4-vrrp1] service vrrp
[DeviceA-security-policy-ip-4-vrrp1] action pass
[DeviceA-security-policy-ip-4-vrrp1] quit
[DeviceA-security-policy-ip] rule name vrrp2
[DeviceA-security-policy-ip-5-vrrp2] source-zone local
[DeviceA-security-policy-ip-5-vrrp2] destination-zone trust
[DeviceA-security-policy-ip-5-vrrp2] service vrrp
[DeviceA-security-policy-ip-5-vrrp2] action pass
[DeviceA-security-policy-ip-5-vrrp2] quit
[DeviceA-security-policy-ip] rule name vrrp3
[DeviceA-security-policy-ip-6-vrrp3] source-zone untrust
[DeviceA-security-policy-ip-6-vrrp3] destination-zone local
[DeviceA-security-policy-ip-6-vrrp3] service vrrp
[DeviceA-security-policy-ip-6-vrrp3] action pass
[DeviceA-security-policy-ip-6-vrrp3] quit
[DeviceA-security-policy-ip] rule name vrrp4
[DeviceA-security-policy-ip-7-vrrp4] source-zone local
[DeviceA-security-policy-ip-7-vrrp4] destination-zone untrust
[DeviceA-security-policy-ip-7-vrrp4] service vrrp
[DeviceA-security-policy-ip-7-vrrp4] action pass
[DeviceA-security-policy-ip-7-vrrp4] quit
[DeviceA-security-policy-ip] quit
```

e. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ip 10.2.1.2
[DeviceA-remote-backup-group] local-ip 10.2.1.1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] backup-mode dual-active
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] delay-time 1
RBM_P[DeviceA-remote-backup-group] quit
```

# Create VRRP groups and associate them with the hot backup system.

```
RBM_P[DeviceA] interface gigabitethernet 1/0/1
RBM_P[DeviceA-GigabitEthernet1/0/1] vrrp vrid 1 virtual-ip 2.1.1.3 active
RBM_P[DeviceA-GigabitEthernet1/0/1] vrrp vrid 2 virtual-ip 2.1.1.4 standby
RBM_P[DeviceA-GigabitEthernet1/0/1] quit
RBM_P[DeviceA] interface gigabitethernet 1/0/2
RBM_P[DeviceA-GigabitEthernet1/0/2] vrrp vrid 3 virtual-ip 10.1.1.3 active
RBM_P[DeviceA-GigabitEthernet1/0/2] vrrp vrid 4 virtual-ip 10.1.1.4 standby
```

```
RBM_P[DeviceA-GigabitEthernet1/0/2] quit
```

   f. Configure security services on Device A. (Details not shown.)

6. Configure Device B:

   a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.1.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

   c. Configure settings for routing. This example configures a static route, and the next hop in the route is 2.1.1.15.

```
[DeviceB] ip route-static 0.0.0.0 0.0.0.0 2.1.1.15
```

   d. Configure hot backup settings.

   # Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ip 10.2.1.1
[DeviceB-remote-backup-group] local-ip 10.2.1.2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] backup-mode dual-active
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
RBM_S[DeviceB-remote-backup-group] delay-time 1
RBM_S[DeviceB-remote-backup-group] quit
```

   # Create VRRP groups and associate them with the hot backup system.

```
RBM_S[DeviceB] interface gigabitethernet 1/0/1
RBM_S[DeviceB-GigabitEthernet1/0/1] vrrp vrid 1 virtual-ip 2.1.1.3 standby
RBM_S[DeviceB-GigabitEthernet1/0/1] vrrp vrid 2 virtual-ip 2.1.1.4 active
RBM_S[DeviceB-GigabitEthernet1/0/1] quit
RBM_S[DeviceB] interface gigabitethernet 1/0/2
RBM_S[DeviceB-GigabitEthernet1/0/2] vrrp vrid 3 virtual-ip 10.1.1.3 standby
RBM_S[DeviceB-GigabitEthernet1/0/2] vrrp vrid 4 virtual-ip 10.1.1.4 active
RBM_S[DeviceB-GigabitEthernet1/0/2] quit
```

7. On some hosts, specify 10.1.1.3 (virtual IP address of VRRP group 3) as the default gateway. On the other hosts, specify 10.1.1.4 (virtual IP address of VRRP group 4) as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

   # Verify that the hot backup channels have been set up.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that Device A is the master in VRRP groups 1 and 3 and the backup in VRRP groups 2 and 4.

```
RBM_P[DeviceA] display vrrp
IPv4 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   VRRP active group status : Master
   VRRP standby group status: Backup
 Total number of virtual routers : 4
 Interface          VRID  State        Running Adver   Auth     Virtual
                                       Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Master       100     100     None     2.1.1.3
 GE1/0/1            2     Backup       100     100     None     2.1.1.4
 GE1/0/2            3     Master       100     100     None     10.1.1.3
 GE1/0/2            4     Backup       100     100     None     10.1.1.4
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device A generates log messages when a host for which Device A forwards traffic communicates with the Internet. Verity that Device A does not generate log messages when a host for which Device B forwards traffic communicates with the Internet. (Details not shown.)

2. Verify the configuration on Device B:

# Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Secondary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that Device B is the master in VRRP groups 2 and 4 and the backup in VRRP groups 1 and 3.

```
RBM_S[DeviceB] display vrrp
IPv4 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   VRRP active group status : Master
   VRRP standby group status: Backup
 Total number of virtual routers : 4
 Interface          VRID  State        Running Adver   Auth     Virtual
                                       Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Backup       100     100     None     2.1.1.3
 GE1/0/1            2     Master       100     100     None     2.1.1.4
 GE1/0/2            3     Backup       100     100     None     10.1.1.3
 GE1/0/2            4     Master       100     100     None     10.1.1.4
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device B generates log messages when a host for which Device B forwards traffic communicates with the Internet. Verity that Device B does not generate log messages when a host for which Device A forwards traffic communicates with the Internet. (Details not shown.)

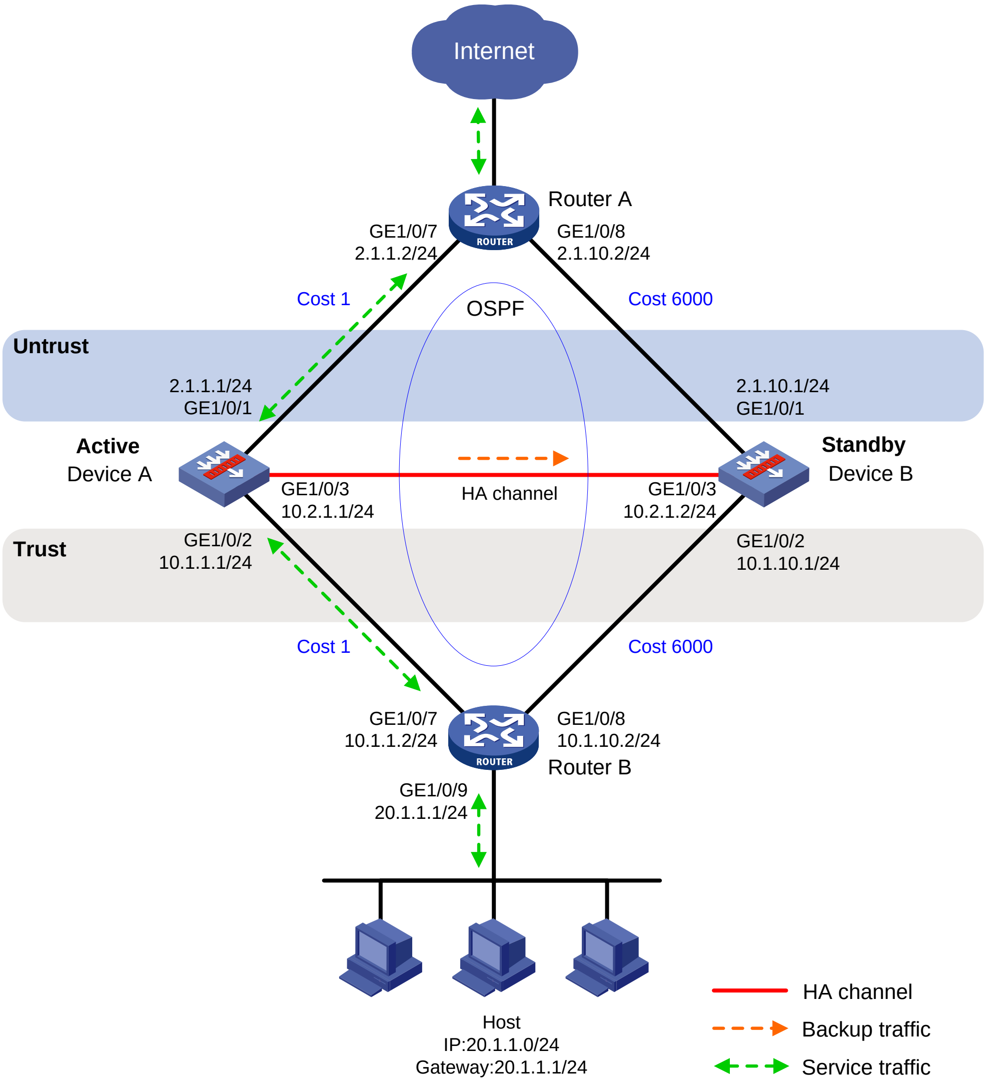
Example: Configuring active/standby hot backup in collaboration with a routing protocol

Network configuration

As shown in Figure 21, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with OSPF.
- Configure the hot backup system to operate in active/standby mode.
- Configure Device A and Device B as the primary device and the secondary device, respectively.

Figure 21 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Router A:

   NOTE:

   This step only provides the brief configuration procedure.

   # Assign 2.1.1.2/24 to GigabitEthernet 1/0/7.

   # Assign 2.1.10.2/24 to GigabitEthernet 1/0/8.

   # Configure OSPF for Device A, Device B, and the routers to have Layer 3 reachability.
3. Configure Router B:

   NOTE:

   This step only provides the brief configuration procedure.

   # Assign 10.1.1.2/24 to GigabitEthernet 1/0/7.

   # Assign 10.1.10.2/24 to GigabitEthernet 1/0/8.

# Configure OSPF for Device A, Device B, and the routers to have Layer 3 reachability.

4. Configure Device A:

   a. Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 2.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   ```

   c. Configure OSPF. Use the default OSPF link cost configuration.

   ```
   [DeviceA] router id 2.1.1.1
   [DeviceA] ospf
   [DeviceA-ospf-1] area 0
   [DeviceA-ospf-1-area-0.0.0.0] network 2.1.1.0 0.0.0.255
   [DeviceA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
   [DeviceA-ospf-1-area-0.0.0.0] quit
   [DeviceA-ospf-1] quit
   ```

   d. Configure a security policy.

   Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

   # Configure a rule named **trust-untrust** to permit the packets from 20.1.1.0/24 to the Internet.

   ```
   [DeviceA] security-policy ip
   [DeviceA-security-policy-ip] rule name trust-untrust
   [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
   [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
   [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 20.1.1.0 24
   [DeviceA-security-policy-ip-3-trust-untrust] action pass
   [DeviceA-security-policy-ip-3-trust-untrust] quit
   ```

   # Configure rules to permit OSPF protocol packets.

   ```
   [DeviceA-security-policy-ip] rule name ospf1
   [DeviceA-security-policy-ip-4-ospf1] source-zone trust
   [DeviceA-security-policy-ip-4-ospf1] destination-zone local
   [DeviceA-security-policy-ip-4-ospf1] service ospf
   [DeviceA-security-policy-ip-4-ospf1] action pass
   [DeviceA-security-policy-ip-4-ospf1] quit
   [DeviceA-security-policy-ip] rule name ospf2
   [DeviceA-security-policy-ip-5-ospf2] source-zone local
   [DeviceA-security-policy-ip-5-ospf2] destination-zone trust
   [DeviceA-security-policy-ip-5-ospf2] service ospf
   [DeviceA-security-policy-ip-5-ospf2] action pass
   ```

```
[DeviceA-security-policy-ip-5-ospf2] quit
[DeviceA-security-policy-ip] rule name ospf3
[DeviceA-security-policy-ip-6-ospf3] source-zone untrust
[DeviceA-security-policy-ip-6-ospf3] destination-zone local
[DeviceA-security-policy-ip-6-ospf3] service ospf
[DeviceA-security-policy-ip-6-ospf3] action pass
[DeviceA-security-policy-ip-6-ospf3] quit
[DeviceA-security-policy-ip] rule name ospf4
[DeviceA-security-policy-ip-7-ospf4] source-zone local
[DeviceA-security-policy-ip-7-ospf4] destination-zone untrust
[DeviceA-security-policy-ip-7-ospf4] service ospf
[DeviceA-security-policy-ip-7-ospf4] action pass
[DeviceA-security-policy-ip-7-ospf4] quit
[DeviceA-security-policy-ip] quit
```

e. Configure hot backup settings.

# Associate track entries with interfaces.

```
[DeviceA] track 1 interface gigabitethernet 1/0/1
[DeviceA-track-1] quit
[DeviceA] track 2 interface gigabitethernet 1/0/2
[DeviceA-track-2] quit
```

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ip 10.2.1.2
[DeviceA-remote-backup-group] local-ip 10.2.1.1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] undo backup-mode
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
```

# Configure the hot backup system to change the link costs advertised in OSPF routes to 6000.

```
RBM_P[DeviceA-remote-backup-group] adjust-cost ospf enable absolute 6000
```

# Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_P[DeviceA-remote-backup-group] track 1
RBM_P[DeviceA-remote-backup-group] track 2
RBM_P[DeviceA-remote-backup-group] quit
```

f. Configure security services on Device A. (Details not shown.)

5. Configure Device B:

a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.1.10.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
```

```
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

c. Configure OSPF. Use the default OSPF link cost configuration.

```
[DeviceB] router id 2.1.10.1
[DeviceB] ospf
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 2.1.10.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] network 10.1.10.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] quit
```

# Associate track entries with interfaces.

```
[DeviceB] track 1 interface gigabitethernet 1/0/1
[DeviceB-track-1] quit
[DeviceB] track 2 interface gigabitethernet 1/0/2
[DeviceB-track-2] quit
```

# Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ip 10.2.1.1
[DeviceB-remote-backup-group] local-ip 10.2.1.2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] undo backup-mode
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
```

# Configure the hot backup system to change the link costs advertised in OSPF routes to 6000.

```
RBM_S[DeviceB-remote-backup-group] adjust-cost ospf enable absolute 6000
```

# Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_S[DeviceB-remote-backup-group] track 1
RBM_S[DeviceB-remote-backup-group] track 2
RBM_S[DeviceB-remote-backup-group] quit
```

6. On the host, specify 20.1.1.1 as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

# Verify that the hot backup channels have been set up.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that the OSPF routes advertised by Device A include a smaller link cost than that advertised by Device B.

```
RBM_P[DeviceA] display ospf interface

         OSPF Process 1 with Router ID 2.1.1.1
                 Interfaces

 Area: 0.0.0.0
 IP Address      Type      State    Cost  Pri   DR              BDR
 2.1.1.1         Broadcast BDR      1     1     2.1.1.2         2.1.1.1
 10.1.1.1        Broadcast DR       1     1     10.1.1.1        10.1.1.2
```

2. Verify the configuration on Device B:

# Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Secondary
  Device running status: Standby
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that the OSPF routes advertised by Device B include a larger link cost than that advertised by Device A.

```
RBM_S[DeviceB] display ospf interface
```

```
          OSPF Process 1 with Router ID 2.1.10.1
                  Interfaces

 Area: 0.0.0.0
 IP Address      Type      State    Cost  Pri   DR              BDR
 2.1.10.1        Broadcast BDR      6000  1     2.1.10.2        2.1.10.1
 10.1.10.1       Broadcast BDR      6000  1     10.1.10.2       10.1.10.1
```

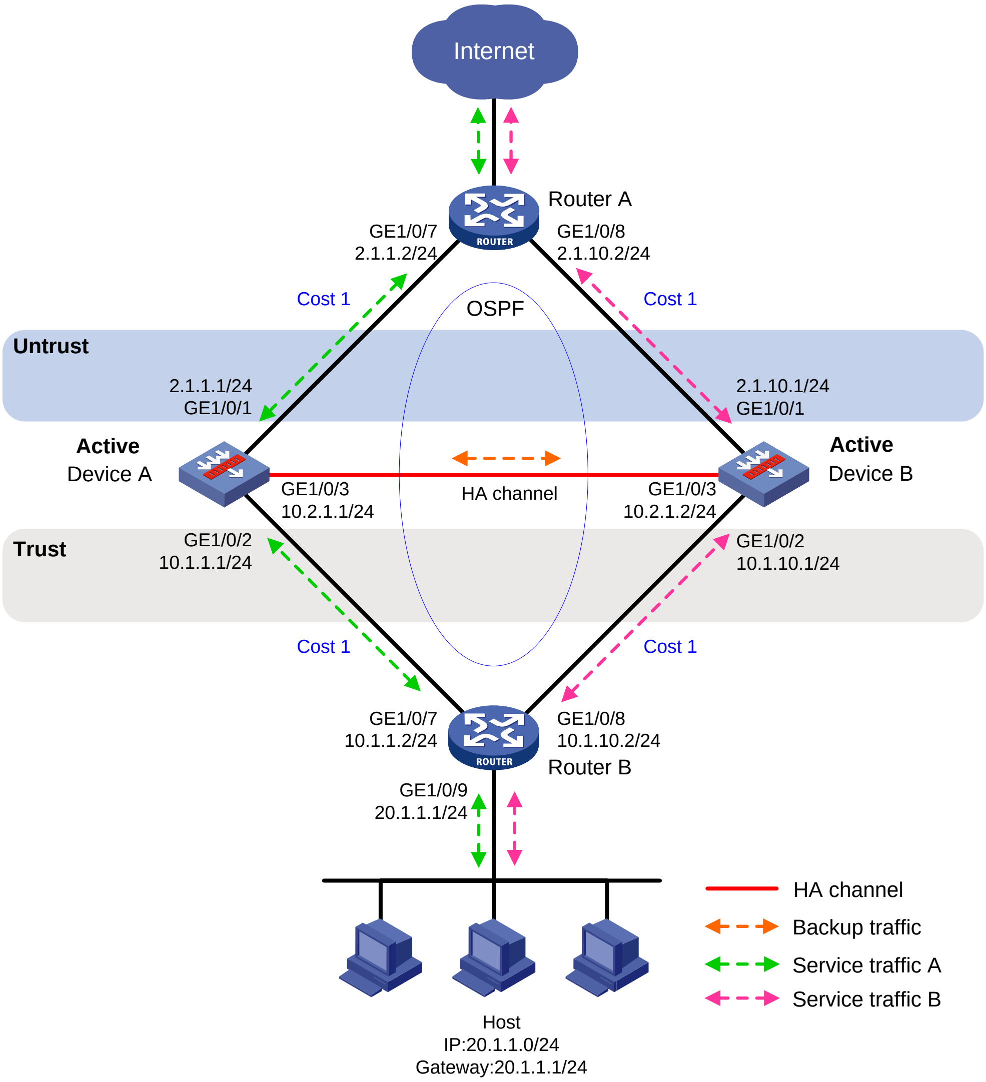
Example: Configuring dual-active hot backup in collaboration with a routing protocol

Network configuration

As shown in Figure 22, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with OSPF.
- Configure the hot backup system to operate in dual-active mode.
- Configure Device A and Device B to load share traffic.

Figure 22 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Router A:

NOTE:

This step only provides the brief configuration procedure.

# Assign 2.1.1.2/24 to GigabitEthernet 1/0/7.

# Assign 2.1.10.2/24 to GigabitEthernet 1/0/8.

# Configure OSPF for Device A, Device B, and the routers to have Layer 3 reachability.

# Configure per-flow load sharing for IP forwarding.

3. Configure Router B:

NOTE:

This step only provides the brief configuration procedure.

# Assign 10.1.1.2/24 to GigabitEthernet 1/0/7.

# Assign 10.1.10.2/24 to GigabitEthernet 1/0/8.

# Configure OSPF for Device A, Device B, and the routers to have Layer 3 reachability.

# Configure per-flow load sharing for IP forwarding.

4. Configure Device A:

   a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 2.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

   c. Configure OSPF. Use the default OSPF link cost configuration.

```
[DeviceA] router id 2.1.1.1
[DeviceA] ospf
[DeviceA-ospf-1] area 0
[DeviceA-ospf-1-area-0.0.0.0] network 2.1.1.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.0] quit
[DeviceA-ospf-1] quit
```

   d. Configure a security policy.

   Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

   # Configure a rule named **trust-untrust** to permit the packets from 20.1.1.0/24 to the Internet.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-3-trust-untrust] action pass
[DeviceA-security-policy-ip-3-trust-untrust] quit
```

   # Configure rules to permit OSPF protocol packets.

```
[DeviceA-security-policy-ip] rule name ospf1
[DeviceA-security-policy-ip-4-ospf1] source-zone trust
[DeviceA-security-policy-ip-4-ospf1] destination-zone local
[DeviceA-security-policy-ip-4-ospf1] service ospf
[DeviceA-security-policy-ip-4-ospf1] action pass
[DeviceA-security-policy-ip-4-ospf1] quit
[DeviceA-security-policy-ip] rule name ospf2
[DeviceA-security-policy-ip-5-ospf2] source-zone local
[DeviceA-security-policy-ip-5-ospf2] destination-zone trust
```

```
[DeviceA-security-policy-ip-5-ospf2] service ospf
[DeviceA-security-policy-ip-5-ospf2] action pass
[DeviceA-security-policy-ip-5-ospf2] quit
[DeviceA-security-policy-ip] rule name ospf3
[DeviceA-security-policy-ip-6-ospf3] source-zone untrust
[DeviceA-security-policy-ip-6-ospf3] destination-zone local
[DeviceA-security-policy-ip-6-ospf3] service ospf
[DeviceA-security-policy-ip-6-ospf3] action pass
[DeviceA-security-policy-ip-6-ospf3] quit
[DeviceA-security-policy-ip] rule name ospf4
[DeviceA-security-policy-ip-7-ospf4] source-zone local
[DeviceA-security-policy-ip-7-ospf4] destination-zone untrust
[DeviceA-security-policy-ip-7-ospf4] service ospf
[DeviceA-security-policy-ip-7-ospf4] action pass
[DeviceA-security-policy-ip-7-ospf4] quit
[DeviceA-security-policy-ip] quit
```

e. Configure hot backup settings.

# Associate track entries with interfaces.

```
[DeviceA] track 1 interface gigabitethernet 1/0/1
[DeviceA-track-1] quit
[DeviceA] track 2 interface gigabitethernet 1/0/2
[DeviceA-track-2] quit
```

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ip 10.2.1.2
[DeviceA-remote-backup-group] local-ip 10.2.1.1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] backup-mode dual-active
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] delay-time 1
```

# Configure the hot backup system to change the link costs advertised in OSPF routes to 1.

```
RBM_P[DeviceA-remote-backup-group] adjust-cost ospf enable absolute 1
```

# Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_P[DeviceA-remote-backup-group] track 1
RBM_P[DeviceA-remote-backup-group] track 2
RBM_P[DeviceA-remote-backup-group] quit
```

f. Configure security services on Device A. (Details not shown.)

5. Configure Device B:

a. Assign an IP address to GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 2.1.10.1 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

c. Configure OSPF. Use the default OSPF link cost configuration.

```
[DeviceB] router id 2.1.10.1
[DeviceB] ospf
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 2.1.10.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] network 10.1.10.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] quit
```

d. Configure hot backup settings.

# Associate track entries with interfaces.

```
[DeviceB] track 1 interface gigabitethernet 1/0/1
[DeviceB-track-1] quit
[DeviceB] track 2 interface gigabitethernet 1/0/2
[DeviceB-track-2] quit
```

# Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ip 10.2.1.1
[DeviceB-remote-backup-group] local-ip 10.2.1.2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] backup-mode dual-active
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
RBM_S[DeviceB-remote-backup-group] delay-time 1
```

# Configure the hot backup system to change the link costs advertised in OSPF routes to 1.

```
RBM_S[DeviceB-remote-backup-group] adjust-cost ospf enable absolute 1
```

# Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_S[DeviceB-remote-backup-group] track 1
RBM_S[DeviceB-remote-backup-group] track 2
RBM_S[DeviceB-remote-backup-group] quit
```

6. On the hosts, specify 20.1.1.1 as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

# Verify that the hot backup channels have been set up.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that the OSPF routes advertised by Device A and Device B include the same link cost.

```
RBM_P[DeviceA] display ospf interface

         OSPF Process 1 with Router ID 2.1.1.1
                 Interfaces

 Area: 0.0.0.0
 IP Address      Type      State    Cost  Pri   DR              BDR
 2.1.1.1         Broadcast BDR      1     1     2.1.1.2         2.1.1.1
 10.1.1.1        Broadcast DR       1     1     10.1.1.1        10.1.1.2
```

2. Verify the configuration on Device B:

# Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Secondary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that the OSPF routes advertised by Device A and Device B include the same link cost.

```
RBM_S[DeviceB] display ospf interface

         OSPF Process 1 with Router ID 2.1.10.1
                 Interfaces

 Area: 0.0.0.0
 IP Address      Type      State    Cost  Pri   DR              BDR
 2.1.10.1        Broadcast BDR      1     1     2.1.10.2        2.1.10.1
 10.1.10.1       Broadcast BDR      1     1     10.1.10.2       10.1.10.1
```

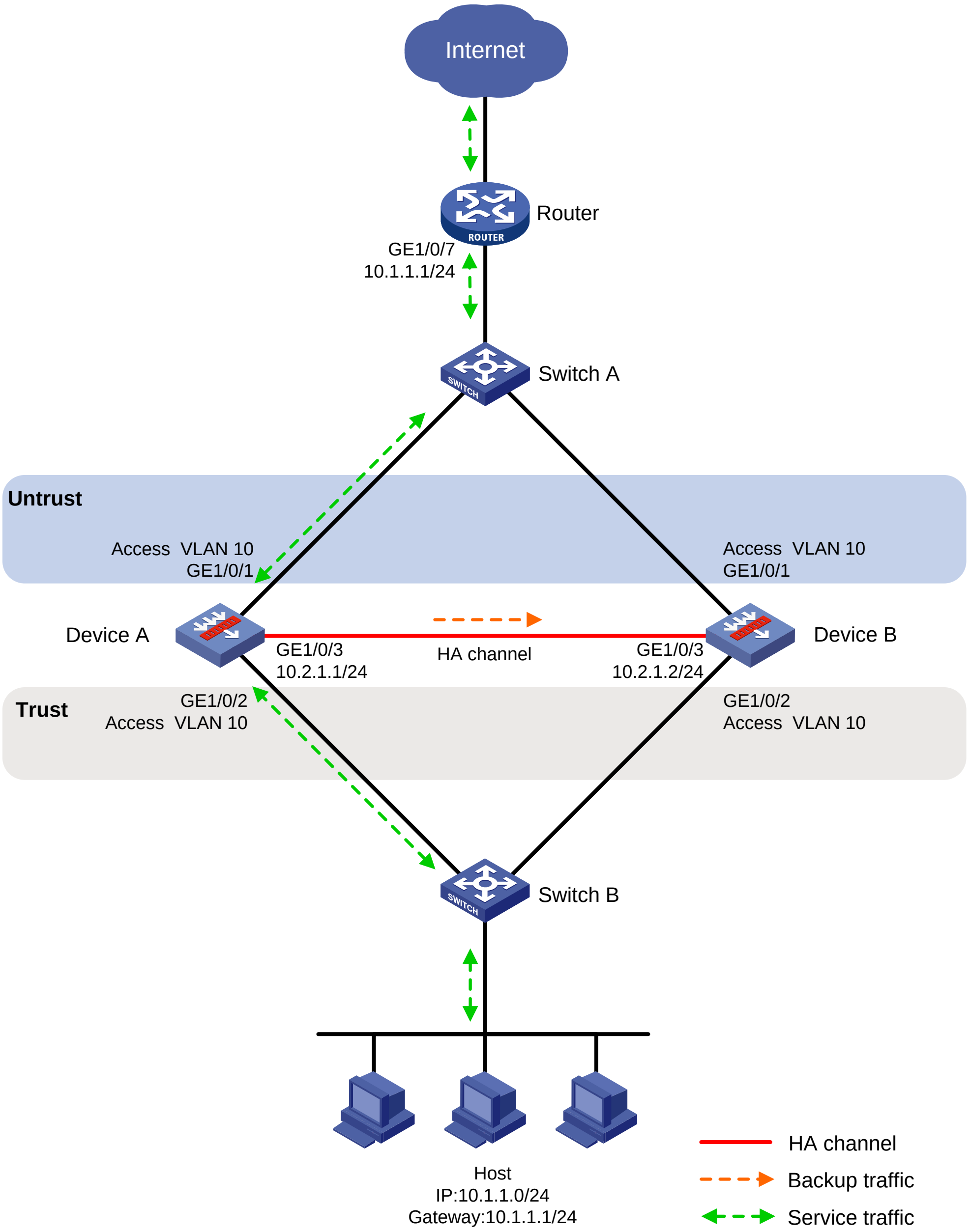
Example: Configuring a transparent in-path hot backup system in active/standby mode

Network configuration

As shown in Figure 23, set up a transparent in-path hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to operate in active/standby mode.
- Connect Switch A and Switch B to Layer 2 interfaces of the hot backup system.
- Configure Device A and Device B as the primary device and the secondary device, respectively.

Figure 23 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Switch A:

   NOTE:

   This step only provides the brief configuration procedure.

   # Create VLAN 10.

   # Configure the interfaces attached to the hot backup system and the router to operate at Layer 2. Assign them to VLAN 10 as access interfaces.
3. Configure Switch B:

NOTE:

This step only provides the brief configuration procedure.

# Create VLAN 10.

# Configure the interfaces attached to the hot backup system and the host to operate at Layer 2. Assign them to VLAN 10 as access interfaces.

4. Configure Device A:

   a. Configure VLANs.

      # Configure the uplink and downlink interfaces to operate at Layer 2.

      ```
      <DeviceA> system-view
      [DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/2
      [DeviceA-if-range] port link-mode bridge
      [DeviceA-if-range] quit
      ```

      # Assign the uplink and downlink interfaces to VLAN 10.

      ```
      [DeviceA] vlan 10
      [DeviceA-vlan10] port gigabitethernet 1/0/1
      [DeviceA-vlan10] port gigabitethernet 1/0/2
      [DeviceA-vlan10] quit
      ```

   b. Assign an IP address to GigabitEthernet 1/0/3.

      ```
      [DeviceA] interface gigabitethernet 1/0/3
      [DeviceA-GigabitEthernet1/0/3] ip address 10.2.1.1 255.255.255.0
      [DeviceA-GigabitEthernet1/0/3] quit
      ```

      # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   c. Add interfaces to security zones.

      ```
      [DeviceA] security-zone name untrust
      [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1 vlan 10
      [DeviceA-security-zone-Untrust] quit
      [DeviceA] security-zone name trust
      [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2 vlan 10
      [DeviceA-security-zone-Trust] quit
      ```

   d. Configure a rule named **trust-untrust** to permit the packets from 10.1.1.0/24 to the Internet.

      Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      [DeviceA-security-policy-ip] quit
      ```

   e. Configure hot backup settings.

      # Set up a hot backup system.

      ```
      [DeviceA] remote-backup group
      [DeviceA-remote-backup-group] remote-ip 10.2.1.2
      [DeviceA-remote-backup-group] local-ip 10.2.1.1
      [DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
      ```

```
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] undo backup-mode
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
```

# Configure the hot backup system to monitor the status of VLAN 10.

```
RBM_P[DeviceA-remote-backup-group] track vlan 10
RBM_P[DeviceA-remote-backup-group] quit
```

f. Configure security services on Device A. (Details not shown.)

5. Configure Device B:

a. Configure VLANs.

# Configure the uplink and downlink interfaces to operate at Layer 2.

```
<DeviceB> system-view
[DeviceB] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/2
[DeviceB-if-range] port link-mode bridge
[DeviceB-if-range] quit
```

# Assign the uplink and downlink interfaces to VLAN 10.

```
[DeviceB] vlan 10
[DeviceB-vlan10] port gigabitethernet 1/0/1
[DeviceB-vlan10] port gigabitethernet 1/0/2
[DeviceB-vlan10] quit
```

b. Assign an IP address to GigabitEthernet 1/0/3.

```
[DeviceB] interface gigabitethernet 1/0/3
[DeviceB-GigabitEthernet1/0/3] ip address 10.2.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/3] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

c. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1 vlan 10
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2 vlan 10
[DeviceB-security-zone-Trust] quit
```

d. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ip 10.2.1.1
[DeviceB-remote-backup-group] local-ip 10.2.1.2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] undo backup-mode
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
```

# Configure the hot backup system to monitor the state of VLAN 10.

```
RBM_S[DeviceB-remote-backup-group] track vlan 10
RBM_S[DeviceB-remote-backup-group] quit
```

6. On the host, specify 10.1.1.1 as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify that the hot backup channels have been set up on Device A.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

2. Verify that the hot backup channels have been set up on Device B.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Secondary
  Device running status: Standby
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

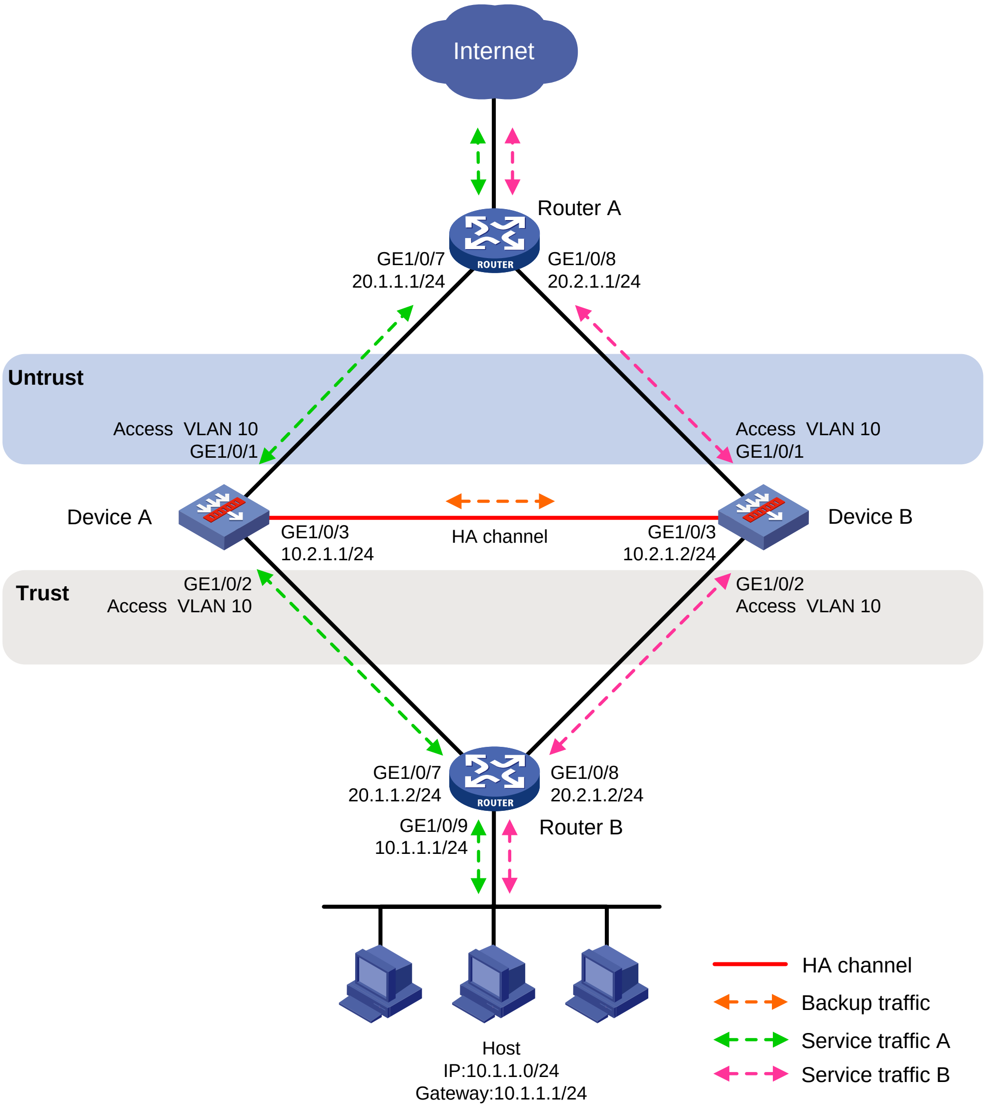
Example: Configuring a transparent in-path hot backup system in dual-active mode

Network configuration

As shown in Figure 24, set up a transparent in-path hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to operate in dual-active mode.
- Connect Router A and Router B to Layer 2 interfaces of the hot backup system.
- Configure Device A and Device B to load share traffic.

Figure 24 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure OSPF on Router A for the host to access the Internet and for Device A and Device B to load share the traffic sent to the host.

3. Configure OSPF on Router B for the host to access the Internet and for Device A and Device B to load share the traffic sent to the Internet.

4. Configure Device A:

   a. Configure VLANs.

      # Configure the uplink and downlink interfaces to operate at Layer 2.

      ```
      <DeviceA> system-view
      [DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/2
      [DeviceA-if-range] port link-mode bridge
      [DeviceA-if-range] quit
      ```

      # Assign the uplink and downlink interfaces to VLAN 10.

      ```
      [DeviceA] vlan 10
      [DeviceA-vlan10] port gigabitethernet 1/0/1
      [DeviceA-vlan10] port gigabitethernet 1/0/2
      [DeviceA-vlan10] quit
      ```

   b. Assign an IP address to GigabitEthernet 1/0/3.

      ```
      [DeviceA] interface GigabitEthernet 1/0/3
      [DeviceA-GigabitEthernet1/0/3] ip address 10.2.1.1 255.255.255.0
      [DeviceA-GigabitEthernet1/0/3] quit
      ```

      # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   c. Add interfaces to security zones.

      ```
      [DeviceA] security-zone name untrust
      [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1 vlan 10
      [DeviceA-security-zone-Untrust] quit
      [DeviceA] security-zone name trust
      [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2 vlan 10
      [DeviceA-security-zone-Trust] quit
      ```

   d. Configure a security policy.

      Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

      # Configure a rule named **trust-untrust** to permit the packets from 10.1.1.0/24 to the Internet.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name trust-untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ip-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ip-3-trust-untrust] source-ip-subnet 10.1.1.0 24
      [DeviceA-security-policy-ip-3-trust-untrust] action pass
      [DeviceA-security-policy-ip-3-trust-untrust] quit
      ```

      # Configure rules to permit OSPF protocol packets.

      ```
      [DeviceA-security-policy-ip] rule name ospf1
      [DeviceA-security-policy-ip-4-ospf1] source-zone trust
      [DeviceA-security-policy-ip-4-ospf1] destination-zone untrust
      [DeviceA-security-policy-ip-4-ospf1] service ospf
      [DeviceA-security-policy-ip-4-ospf1] action pass
      [DeviceA-security-policy-ip-4-ospf1] quit
      [DeviceA-security-policy-ip] rule name ospf2
      [DeviceA-security-policy-ip-5-ospf2] source-zone untrust
      ```

```
[DeviceA-security-policy-ip-5-ospf2] destination-zone trust
[DeviceA-security-policy-ip-5-ospf2] service ospf
[DeviceA-security-policy-ip-5-ospf2] action pass
[DeviceA-security-policy-ip-5-ospf2] quit
[DeviceA-security-policy-ip] quit
```

e. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ip 10.2.1.2
[DeviceA-remote-backup-group] local-ip 10.2.1.1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] backup-mode dual-active
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] delay-time 1
```

# Configure the hot backup system to monitor the status of GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
RBM_P[DeviceA-remote-backup-group] track interface gigabitethernet 1/0/1
RBM_P[DeviceA-remote-backup-group] track interface gigabitethernet 1/0/2
RBM_P[DeviceA-remote-backup-group] quit
```

f. Configure security services on Device A. (Details not shown.)

5. Configure Device B:

a. Configure VLANs.

# Configure the uplink and downlink interfaces to operate at Layer 2.

```
<DeviceB> system-view
[DeviceB] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/2
[DeviceB-if-range] port link-mode bridge
[DeviceB-if-range] quit
```

# Assign the uplink and downlink interfaces to VLAN 10.

```
[DeviceB] vlan 10
[DeviceB-vlan10] port gigabitethernet 1/0/1
[DeviceB-vlan10] port gigabitethernet 1/0/2
[DeviceB-vlan10] quit
```

b. Assign an IP address to GigabitEthernet 1/0/3.

```
[DeviceB] interface gigabitethernet 1/0/3
[DeviceB-GigabitEthernet1/0/3] ip address 10.2.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/3] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

c. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1 vlan 10
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2 vlan 10
[DeviceB-security-zone-Trust] quit
```

d. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ip 10.2.1.1
[DeviceB-remote-backup-group] local-ip 10.2.1.2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] backup-mode dual-active
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] delay-time 1
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
```

# Configure the hot backup system to monitor the status of GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2.

```
RBM_S[DeviceB-remote-backup-group] track interface gigabitethernet 1/0/1
RBM_S[DeviceB-remote-backup-group] track interface gigabitethernet 1/0/2
RBM_S[DeviceB-remote-backup-group] quit
```

6. On the host, specify 10.1.1.1 as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify that the hot backup channels have been set up on Device A.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.1
  Remote IP: 10.2.1.2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change       Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

2. Verify that the hot backup channels have been set up on Device B.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Secondary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IP: 10.2.1.2
  Remote IP: 10.2.1.1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                         Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

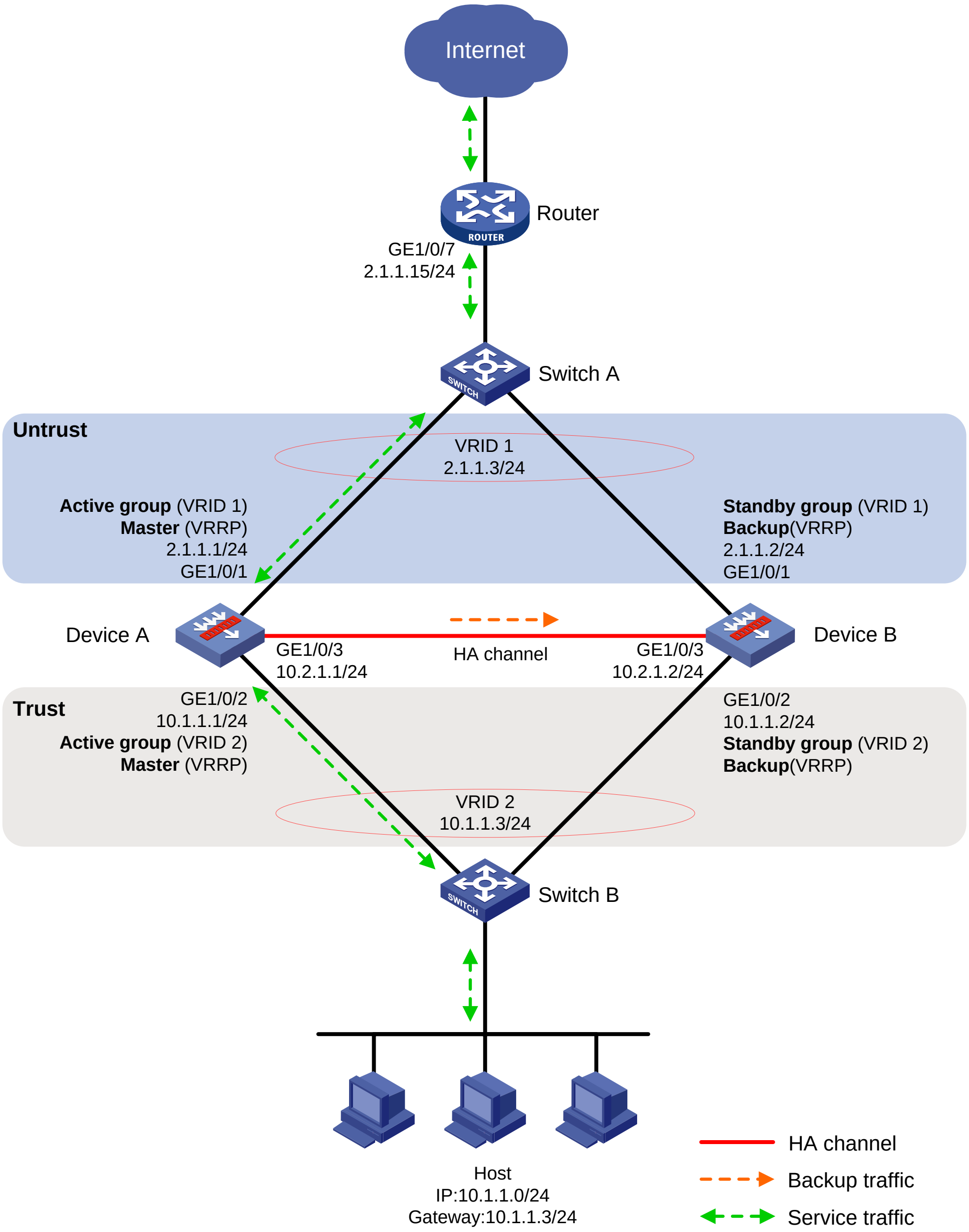
Example: Configuring interface-based NAT on an active/standby hot backup system in collaboration with VRRP

Network configuration

As shown in Figure 25, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.
- Configure the hot backup system to operate in active/standby mode.
- Configure Device A and Device B as the primary device and the secondary device, respectively.
- Configure dynamic NAT to translate the private IP addresses in the internal network into public IP addresses 2.1.1.1 through 2.1.1.10.

Figure 25 Network diagram



Procedure

1. Set up the hot backup system as described in "Example: Configuring active/standby hot backup in collaboration with VRRP."
2. Configure dynamic NAT on Device A (primary):

   # Create NAT address group 1 and add address range 2.1.1.5 to 2.1.1.10. Associate NAT address group 1 with VRRP group 1.

   ```
   RBM_P<DeviceA> system-view
   RBM_P[DeviceA] nat address-group 1
   RBM_P[DeviceA-address-group-1] address 2.1.1.5 2.1.1.10
   RBM_P[DeviceA-address-group-1] vrrp vrid 1
   RBM_P[DeviceA-address-group-1] quit
   ```

# Configure outbound dynamic NAT to use NAT address group 1 for address translation on GigabitEthernet 1/0/1.

```
RBM_P[DeviceA] interface gigabitethernet 1/0/1
RBM_P[DeviceA-GigabitEthernet1/0/1] nat outbound address-group 1
RBM_P[DeviceA-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Verify that the host can communicate with the Internet. (Details not shown.)

# Verify that Device A has generated a NAT session entry.

```
RBM_P[DeviceA] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.10/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.5/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:16:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

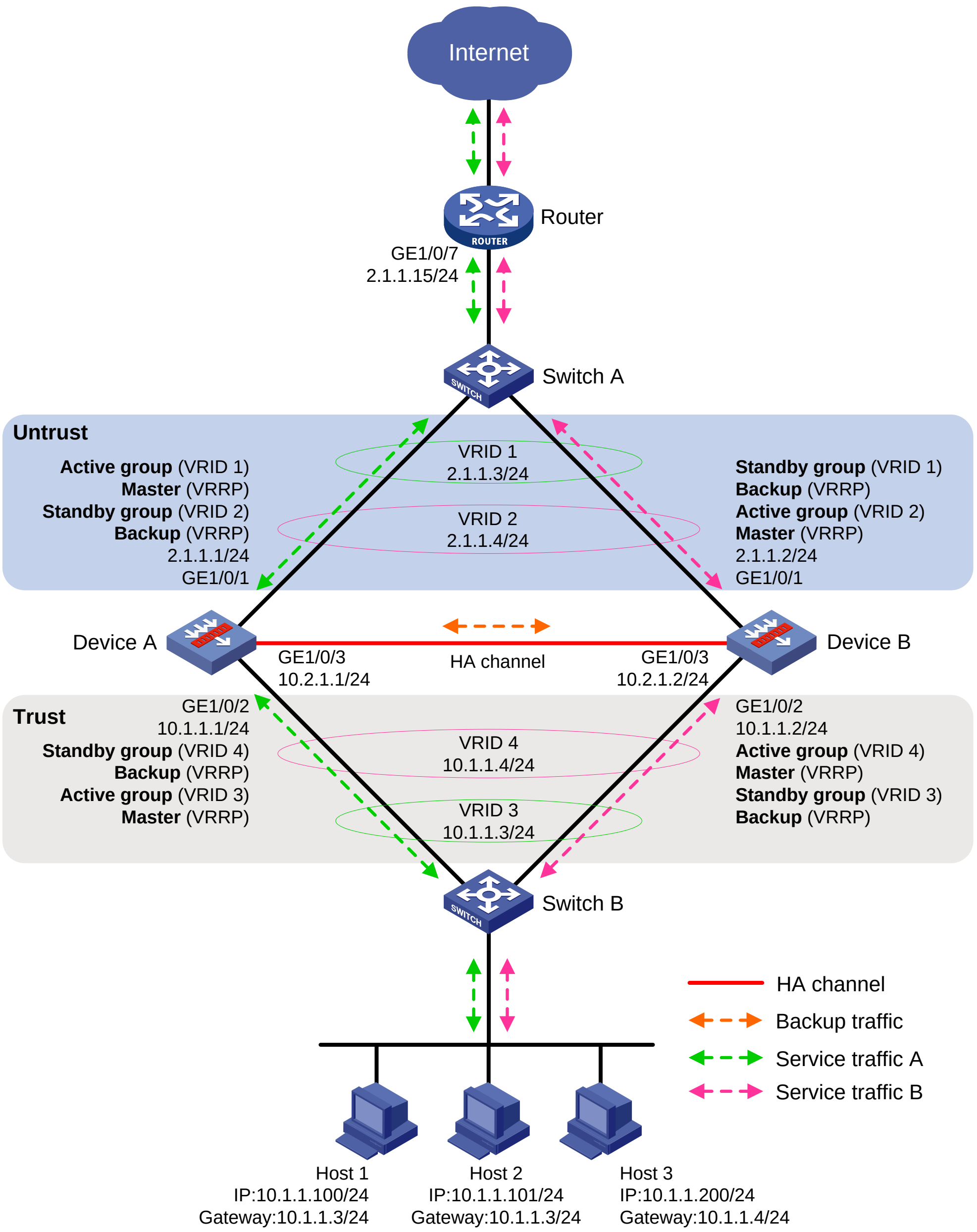
Example: Configuring interface-based NAT on a dual-active hot backup system in collaboration with VRRP

Network configuration

As shown in Figure 26, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.
- Configure the hot backup system to operate in dual-active mode.
- Configure Device A and Device B to load share traffic.
- Configure dynamic NAT to translate the private IP addresses in the internal network into public IP addresses 2.1.1.1 through 2.1.1.10.

Figure 26 Network diagram



Procedure

1. Set up the hot backup system as described in "Example: Configuring dual-active hot backup in collaboration with VRRP".
2. Configure dynamic NAT on Device A (primary):

   # Create NAT address group 1 and add address range 2.1.1.5 to 2.1.1.7. Associate NAT address group 1 with VRRP group 1.

   ```
   RBM_P<DeviceA> system-view
   RBM_P[DeviceA] nat address-group 1
   RBM_P[DeviceA-address-group-1] address 2.1.1.5 2.1.1.7
   RBM_P[DeviceA-address-group-1] vrrp vrid 1
   RBM_P[DeviceA-address-group-1] quit
   ```

   # Create NAT address group 2 and add address range 2.1.1.8 to 2.1.1.10. Associate NAT address group 2 with VRRP group 2.

```
RBM_P[DeviceA] nat address-group 2
RBM_P[DeviceA-address-group-2] address 2.1.1.8 2.1.1.10
RBM_P[DeviceA-address-group-2] vrrp vrid 2
RBM_P[DeviceA-address-group-2] quit
```

# Configure ACL 3000 to permit packets from 10.1.1.1/25. Configure ACL 3001 to permit packets from 10.1.1.129/25.

```
RBM_P[DeviceA] acl advanced 3000
RBM_P[DeviceA-ipv4-adv-3000] rule permit ip source 10.1.1.1 0.0.0.127
RBM_P[DeviceA-ipv4-adv-3000] quit
RBM_P[DeviceA] acl advanced 3001
RBM_P[DeviceA-ipv4-adv-3001] rule permit ip source 10.1.1.129 0.0.0.127
RBM_P[DeviceA-ipv4-adv-3001] quit
```

# Configure outbound dynamic NAT on GigabitEthernet 1/0/1. The source IP addresses of the packets permitted by ACL 3000 are translated into the addresses in NAT address group 1. The source IP addresses of the packets permitted by ACL 3001 are translated into the addresses in NAT address group 2.

```
RBM_P[DeviceA] interface gigabitethernet 1/0/1
RBM_P[DeviceA-GigabitEthernet1/0/1] nat outbound 3000 address-group 1
RBM_P[DeviceA-GigabitEthernet1/0/1] nat outbound 3001 address-group 2
RBM_P[DeviceA-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Verify that Host 1 can communicate with the Internet. (Details not shown.)

# Verify that Device A has generated a NAT session entry.

```
RBM_P[DeviceA] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.100/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.5/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:16:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
```

```
Total sessions found: 1
```

# Verify that Host 3 can communicate with the Internet. (Details not shown.)

# Verify that Device B has generated a NAT session entry.

```
RBM_S[DeviceB] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.200/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface:
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.8/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface:
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:17:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

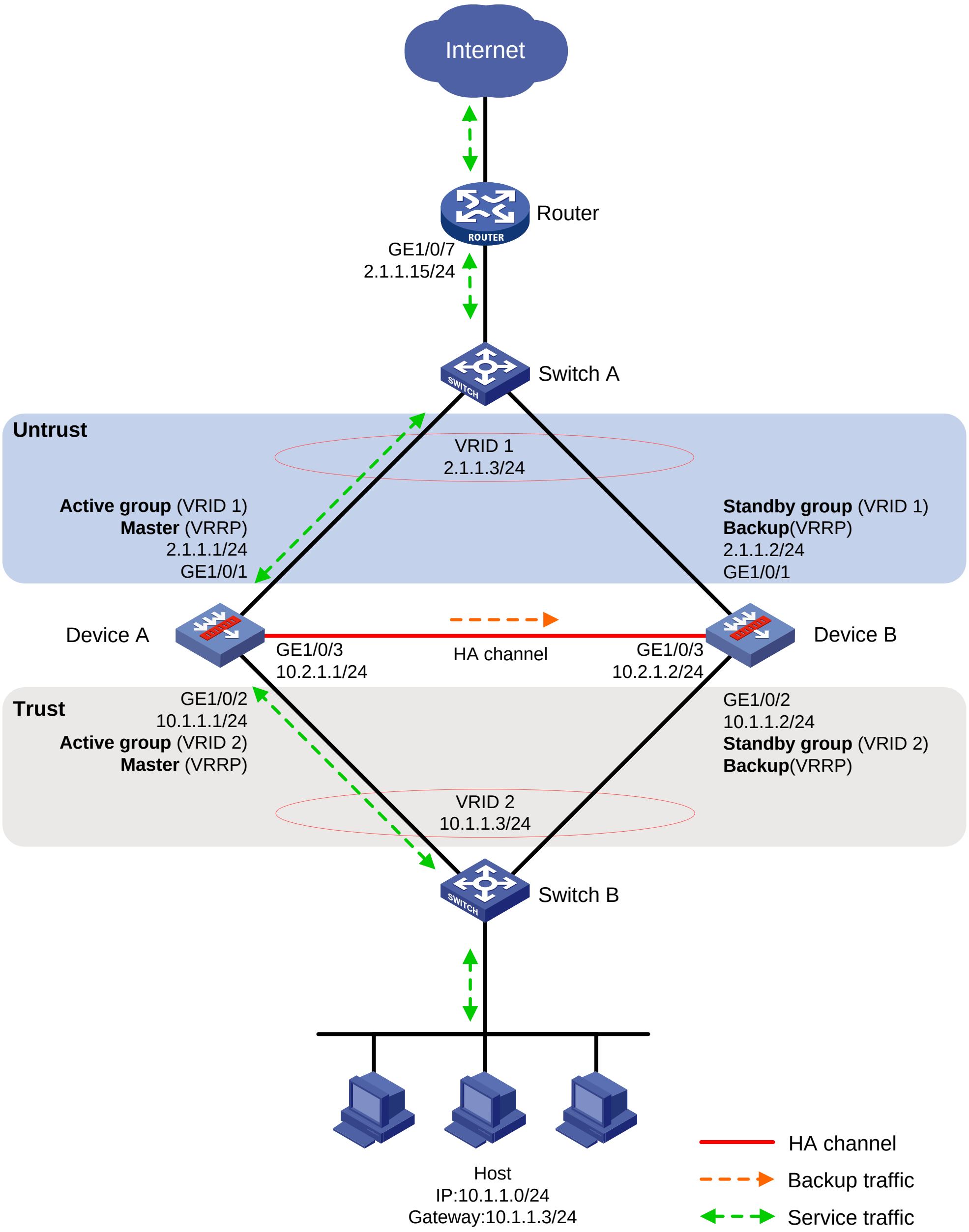
Example: Configuring global NAT on an active/standby hot backup system in collaboration with VRRP

Network configuration

As shown in Figure 27, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.
- Configure the hot backup system to operate in active/standby mode.
- Configure Device A and Device B as the primary device and the secondary device, respectively.
- Configure dynamic NAT to translate the private IP addresses in the internal network into public IP addresses 2.1.1.1 through 2.1.1.10.

Figure 27 Network diagram



Procedure

1. Set up the hot backup system as described in "Example: Configuring active/standby hot backup in collaboration with VRRP."
2. Configure dynamic NAT globally on Device A (primary):

   # Create NAT address group 1 and add address range 2.1.1.5 to 2.1.1.10. Associate NAT address group 1 with VRRP group 1.

```
RBM_P<DeviceA> system-view
RBM_P[DeviceA] nat address-group 1
RBM_P[DeviceA-address-group-1] address 2.1.1.5 2.1.1.10
RBM_P[DeviceA-address-group-1] vrrp vrid 1
RBM_P[DeviceA-address-group-1] quit
```

# Configure a global NAT policy to use NAT address group 1 for address translation of the traffic sent from the internal network to the Internet.

```
RBM_P[DeviceA] nat global-policy
RBM_P[DeviceA-nat-global-policy] rule name rule1
RBM_P[DeviceA-nat-global-policy-rule1] source-zone Trust
RBM_P[DeviceA-nat-global-policy-rule1] destination-zone Untrust
RBM_P[DeviceA-nat-global-policy-rule1] source-ip subnet 10.1.1.0 24
RBM_P[DeviceA-nat-global-policy-rule1] action snat address-group 1 vrrp 1
RBM_P[DeviceA-nat-global-policy-rule1] quit
RBM_P[DeviceA-nat-global-policy] quit
```

Verifying the configuration

# Verify that the host can communicate with the Internet. (Details not shown.)

# Verify that Device A has generated a NAT session entry.

```
RBM_P[DeviceA] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.10/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.5/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:16:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

Example: Configuring global NAT on a dual-active hot backup system in collaboration with VRRP

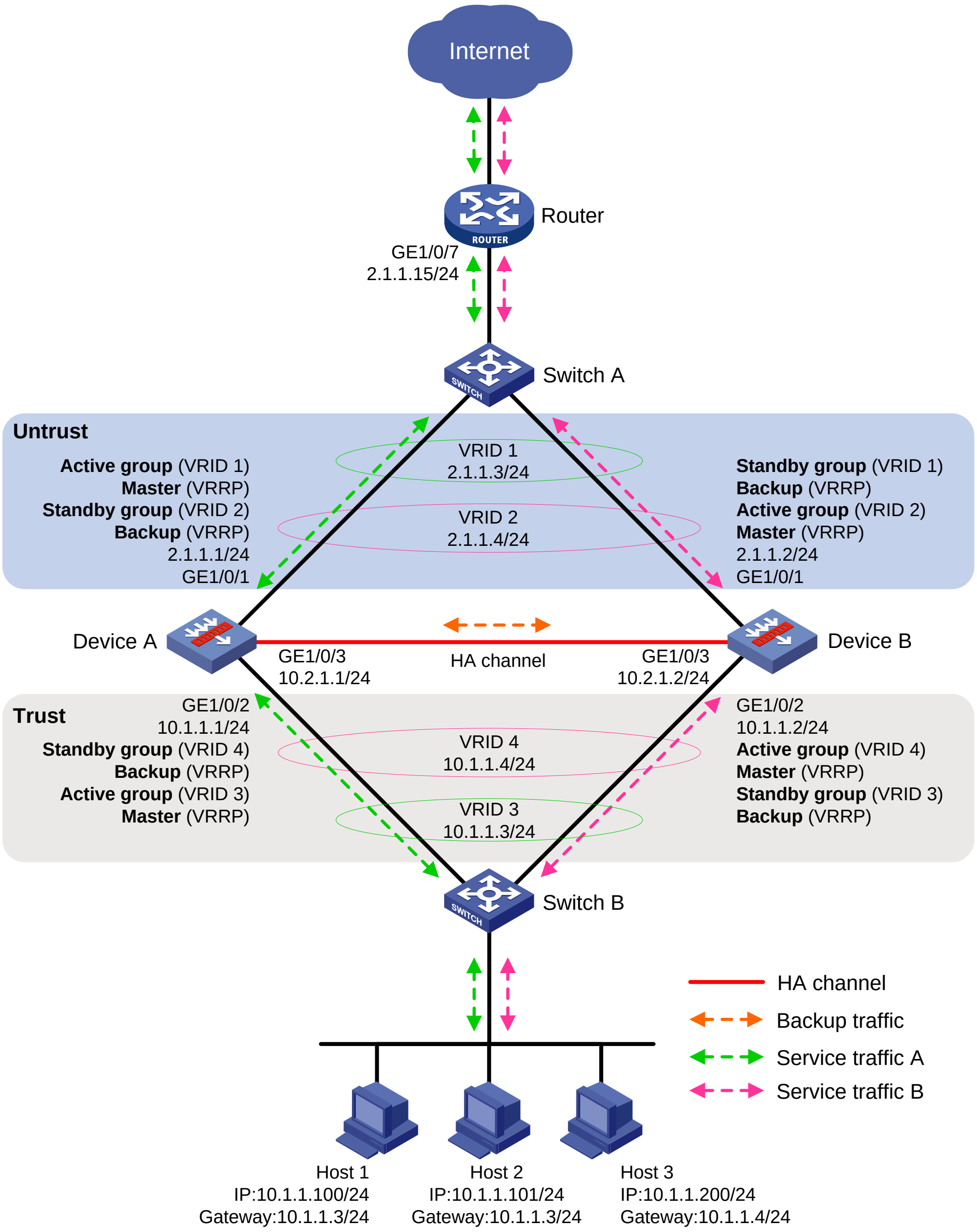
Network configuration

As shown in Figure 28, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.

- Configure the hot backup system to operate in dual-active mode.
- Configure Device A and Device B to load share traffic.
- Configure dynamic NAT to translate the private IP addresses in the internal network into public IP addresses 2.1.1.1 through 2.1.1.10.

Figure 28 Network diagram



Procedure

1. Set up the hot backup system as described in "Example: Configuring dual-active hot backup in collaboration with VRRP".
2. Configure dynamic NAT on Device A (primary):

   # Create NAT address group 1 and add address range 2.1.1.5 to 2.1.1.7. Associate NAT address group 1 with VRRP group 1.

   ```
   RBM_P<DeviceA> system-view
   RBM_P[DeviceA] nat address-group 1
   ```

```
RBM_P[DeviceA-address-group-1] address 2.1.1.5 2.1.1.7
RBM_P[DeviceA-address-group-1] vrrp vrid 1
RBM_P[DeviceA-address-group-1] quit
```

# Create NAT address group 2 and add address range 2.1.1.8 to 2.1.1.10. Associate NAT address group 2 with VRRP group 2.

```
RBM_P[DeviceA] nat address-group 2
RBM_P[DeviceA-address-group-2] address 2.1.1.8 2.1.1.10
RBM_P[DeviceA-address-group-2] vrrp vrid 2
RBM_P[DeviceA-address-group-2] quit
```

# Configure global NAT rule 1 to use NAT address group 1 for address translation of the traffic sent from the 10.1.1.1/25 network to the Internet.

```
RBM_P[DeviceA] nat global-policy
RBM_P[DeviceA-nat-global-policy] rule name rule1
RBM_P[DeviceA-nat-global-policy-rule1] source-zone Trust
RBM_P[DeviceA-nat-global-policy-rule1] destination-zone Untrust
RBM_P[DeviceA-nat-global-policy-rule1] source-ip subnet 10.1.1.1 25
RBM_P[DeviceA-nat-global-policy-rule1] action snat address-group 1 vrrp 1
RBM_P[DeviceA-nat-global-policy-rule1] quit
```

# Configure global NAT rule 2 to use NAT address group 2 for address translation of the traffic sent from the 10.1.1.129/25 network to the Internet.

```
RBM_P[DeviceA-nat-global-policy] rule name rule2
RBM_P[DeviceA-nat-global-policy-rule2] source-zone Trust
RBM_P[DeviceA-nat-global-policy-rule2] destination-zone Untrust
RBM_P[DeviceA-nat-global-policy-rule2] source-ip subnet 10.1.1.129 25
RBM_P[DeviceA-nat-global-policy-rule2] action snat address-group 2 vrrp 2
RBM_P[DeviceA-nat-global-policy-rule2] quit
RBM_P[DeviceA-nat-global-policy] quit
```

Verifying the configuration

# Verify that Host 1 can communicate with the Internet. (Details not shown.)

# Verify that Device A has generated a NAT session entry.

```
RBM_P[DeviceA] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.100/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/2
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.5/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface: GigabitEthernet1/0/1
  Source security zone: Untrust
```

```
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:16:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

# Verify that Host 3 can communicate with the Internet. (Details not shown.)

# Verify that Device B has generated a NAT session entry.

```
RBM_S[DeviceB] display nat session verbose
Slot 1:
Initiator:
  Source      IP/port: 10.1.1.200/52082
  Destination IP/port: 202.38.1.10/80
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface:
  Source security zone: Trust
Responder:
  Source      IP/port: 202.38.1.10/80
  Destination IP/port: 2.1.1.8/1036
  DS-Lite tunnel peer: -
  VPN instance/VLAN ID/Inline ID: -/-/-
  Protocol: TCP(6)
  Inbound interface:
  Source security zone: Untrust
State: TCP_ESTABLISHED
Application: HTTP
Rule ID: 2
Rule name: 3
Start time: 2019-1-29 16:17:59  TTL: 9995s
Initiator->Responder:          551 packets      32547 bytes
Responder->Initiator:          956 packets    1385514 bytes
Total sessions found: 1
```

Hot backup configuration examples (IPv6)

Example: Configuring active/standby hot backup in collaboration with VRRP

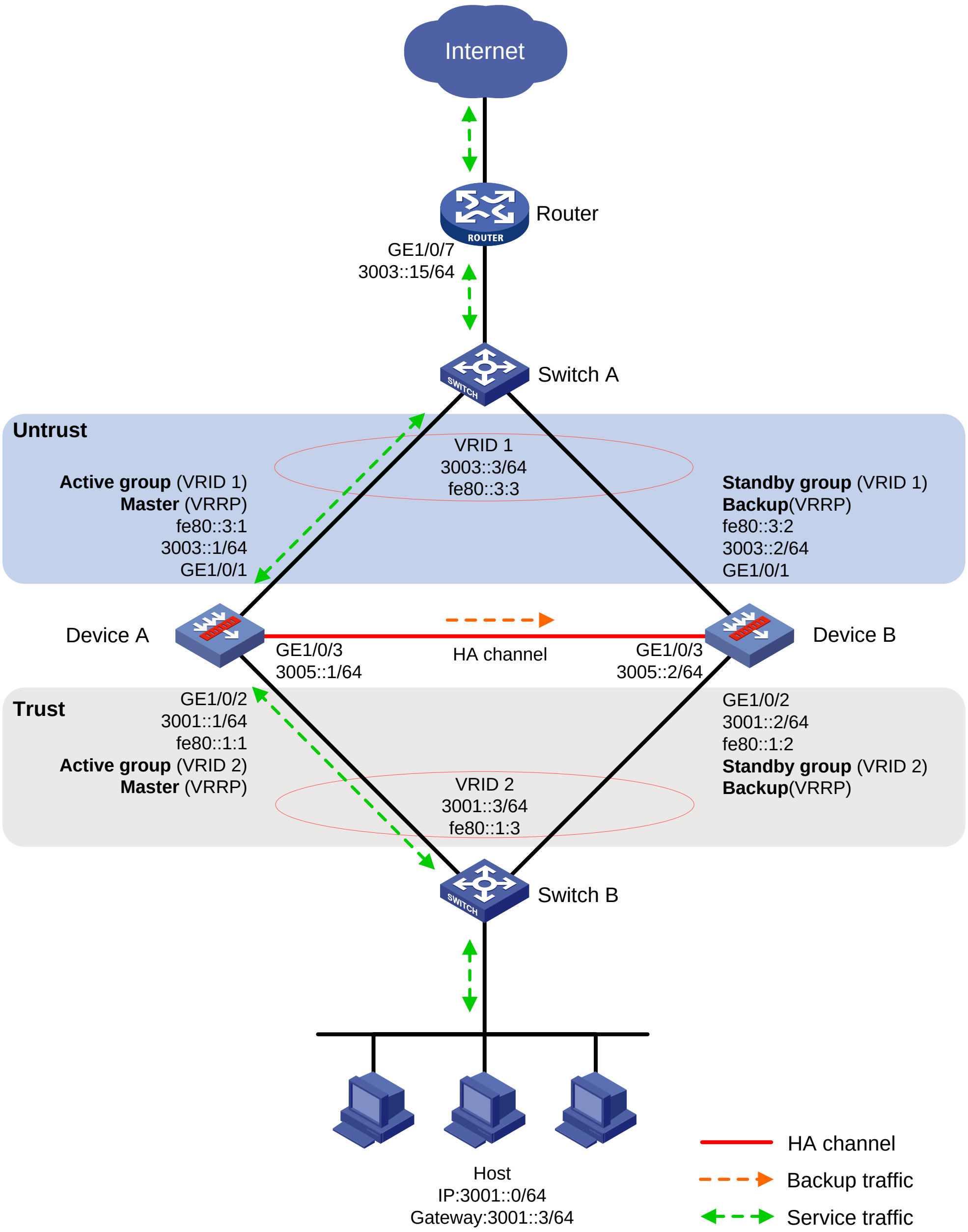
Network configuration

As shown in Figure 19, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with VRRP.

- Configure the hot backup system to operate in active/standby mode.
- Configure Device A and Device B as the primary device and the secondary device, respectively.

Figure 29 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Switch A:

NOTE:

This step only provides the brief configuration procedure.

# Create VLAN 10.

# Configure the interfaces attached to the hot backup system and the router to operate at Layer 2. Assign them to VLAN 10 as access interfaces.

3. Configure Switch B:

NOTE:

This step only provides the brief configuration procedure.

# Create VLAN 10.

# Configure the interfaces attached to the hot backup system and the host to operate at Layer 2. Assign them to VLAN 10 as access interfaces.

4. Configure the router:

NOTE:

This step only provides the brief configuration procedure.

# Assign 3003::15/64 to GigabitEthernet 1/0/7.

# Configure routes as follows:

- Specify 3003::3/64 (virtual IP address of VRRP group 1) as the next hop of the routes to the internal network.
- Specify the IP address of the peer interface attached to the traffic outgoing interface as the next hop of the route to the Internet.

5. Configure Device A:

   a. Assign IP addresses to interfaces.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 3003::1/64
[DeviceA-GigabitEthernet1/0/1] ipv6 address fe80::3:1 link-local
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ipv6 address 3001::1/64
[DeviceA-GigabitEthernet1/0/2] ipv6 address fe80::1:1 link-local
[DeviceA-GigabitEthernet1/0/2] undo ipv6 nd ra halt
[DeviceA-GigabitEthernet1/0/2] quit
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] ipv6 address 3005::1/64
[DeviceA-GigabitEthernet1/0/3] ipv6 address auto link-local
[DeviceA-GigabitEthernet1/0/3] quit
```

   b. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```

   c. Configure settings for routing. This example configures a static route, and the next hop in the route is 3003::15.

```
[DeviceA] ipv6 route-static 0::0 0 3003::15
```

   d. Configure a security policy.

   Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

# Configure security policy rule named **trust-untrust** to permit the packets from 3001::0/64 to the Internet.

```
[DeviceA] security-policy ipv6
[DeviceA-security-policy-ipv6] rule name trust-untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceA-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ipv6-3-trust-untrust] source-ip-subnet 3001::0 64
[DeviceA-security-policy-ipv6-3-trust-untrust] action pass
[DeviceA-security-policy-ipv6-3-trust-untrust] quit
```

# Configure rules to permit VRRP protocol packets. When the hot backup channel is disconnected, Device A and Device B can exchange VRRP protocol packets to elect a VRRP master.

```
[DeviceA-security-policy-ipv6] rule name vrrp1
[DeviceA-security-policy-ipv6-4-vrrp1] source-zone trust
[DeviceA-security-policy-ipv6-4-vrrp1] destination-zone local
[DeviceA-security-policy-ipv6-4-vrrp1] service vrrp
[DeviceA-security-policy-ipv6-4-vrrp1] action pass
[DeviceA-security-policy-ipv6-4-vrrp1] quit
[DeviceA-security-policy-ipv6] rule name vrrp2
[DeviceA-security-policy-ipv6-5-vrrp2] source-zone local
[DeviceA-security-policy-ipv6-5-vrrp2] destination-zone trust
[DeviceA-security-policy-ipv6-5-vrrp2] service vrrp
[DeviceA-security-policy-ipv6-5-vrrp2] action pass
[DeviceA-security-policy-ipv6-5-vrrp2] quit
[DeviceA-security-policy-ipv6] rule name vrrp3
[DeviceA-security-policy-ipv6-6-vrrp3] source-zone untrust
[DeviceA-security-policy-ipv6-6-vrrp3] destination-zone local
[DeviceA-security-policy-ipv6-6-vrrp3] service vrrp
[DeviceA-security-policy-ipv6-6-vrrp3] action pass
[DeviceA-security-policy-ipv6-6-vrrp3] quit
[DeviceA-security-policy-ipv6] rule name vrrp4
[DeviceA-security-policy-ipv6-7-vrrp4] source-zone local
[DeviceA-security-policy-ipv6-7-vrrp4] destination-zone untrust
[DeviceA-security-policy-ipv6-7-vrrp4] service vrrp
[DeviceA-security-policy-ipv6-7-vrrp4] action pass
[DeviceA-security-policy-ipv6-7-vrrp4] quit
[DeviceA-security-policy-ipv6] quit
```

e. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ipv6 3005::2
[DeviceA-remote-backup-group] local-ipv6 3005::1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] undo backup-mode
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] quit
```

# Create VRRP groups and associate them with the hot backup system.

```
RBM_P[DeviceA] interface gigabitethernet 1/0/1
RBM_P[DeviceA-GigabitEthernet1/0/1] vrrp ipv6 vrid 1 virtual-ip fe80::3:3
link-local active
RBM_P[DeviceA-GigabitEthernet1/0/1] vrrp ipv6 vrid 1 virtual-ip 3003::3
RBM_P[DeviceA-GigabitEthernet1/0/1] quit
RBM_P[DeviceA] interface gigabitethernet 1/0/2
RBM_P[DeviceA-GigabitEthernet1/0/2] vrrp ipv6 vrid 1 virtual-ip fe80::1:3
link-local active
RBM_P[DeviceA-GigabitEthernet1/0/2] vrrp ipv6 vrid 1 virtual-ip 3001::3
RBM_P[DeviceA-GigabitEthernet1/0/2] quit
```

f. Configure security services on Device A. (Details not shown.)

6. Configure Device B:

a. Assign IP addresses to interfaces.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 3003::2/64
[DeviceB-GigabitEthernet1/0/1] ipv6 address fe80::3:2 link-local
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ipv6 address 3001::2/64
[DeviceB-GigabitEthernet1/0/2] ipv6 address fe80::1:2 link-local
[DeviceB-GigabitEthernet1/0/2] undo ipv6 nd ra halt
[DeviceB-GigabitEthernet1/0/2] quit
[DeviceB] interface gigabitethernet 1/0/3
[DeviceB-GigabitEthernet1/0/3] ipv6 address 3005::2/64
[DeviceB-GigabitEthernet1/0/3] ipv6 address auto link-local
[DeviceB-GigabitEthernet1/0/3] quit
```

b. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

c. Configure settings for routing. This example configures a static route, and the next hop in the route is 3003::15.

```
[DeviceB] ipv6 route-static 0::0 0 3003::15
```

d. Configure hot backup settings.

# Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ipv6 3005::1
[DeviceB-remote-backup-group] local-ipv6 3005::2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] undo backup-mode
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
```

```
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
RBM_S[DeviceB-remote-backup-group] quit
```

# Create VRRP groups and associate them with the hot backup system.

```
RBM_S[DeviceB] interface gigabitethernet 1/0/1
RBM_S[DeviceB-GigabitEthernet1/0/1] vrrp ipv6 vrid 1 virtual-ip fe80::3:3
link-local standby
RBM_S[DeviceB-GigabitEthernet1/0/1] vrrp ipv6 vrid 1 virtual-ip 3003::3
RBM_S[DeviceB-GigabitEthernet1/0/1] quit
RBM_S[DeviceB] interface gigabitethernet 1/0/2
RBM_S[DeviceB-GigabitEthernet1/0/2] vrrp ipv6 vrid 1 virtual-ip fe80::1:3
link-local standby
RBM_S[DeviceB-GigabitEthernet1/0/2] vrrp ipv6 vrid 1 virtual-ip 3001::3
RBM_S[DeviceB-GigabitEthernet1/0/2] quit
```

7. On the host, specify 3001::3 (virtual IP address of VRRP group 2) as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

# Verify that the hot backup channels have been set up.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that Device A is the master in all VRRP groups.

```
RBM_P[DeviceA] display vrrp ipv6
IPv6 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   IPv6 VRRP active group status : Master
   IPv6 VRRP standby group status: Master
 Total number of virtual routers : 2
 Interface          VRID  State        Running Adver   Auth     Virtual
                                       Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Master       100     100     None     FE80::3:3
 GE1/0/2            1     Master       100     100     None     FE80::1:3
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device A generates log messages when the host communicates with the Internet. (Details not shown.)

2. Verify the configuration on Device B:

# Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Active/standby
  Device management role: Secondary
  Device running status: Standby
  Data channel interface: GigabitEthernet1/0/3
  Local IPv6: 3005::2
  Remote IPv6: 3005::1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 0 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that Device A is the backup in all VRRP groups.

```
RBM_S[DeviceB] display vrrp ipv6
IPv6 Virtual Router Information:
 Running mode      : Standard
 RBM control channel is established
   IPv6 VRRP active group status : Backup
   IPv6 VRRP standby group status: Backup
 Total number of virtual routers : 2
 Interface          VRID  State       Running Adver   Auth     Virtual
                                      Pri     Timer   Type        IP
 ---------------------------------------------------------------------
 GE1/0/1            1     Backup       100     100     None     FE80::3:3
 GE1/0/2            1     Backup       100     100     None     FE80::1:3
```

# Enable logging for the security policy that permits communication between security zones **Trust** and **Untrust**. Verity that Device B does not generate log messages when the host communicates with the Internet. (Details not shown.)

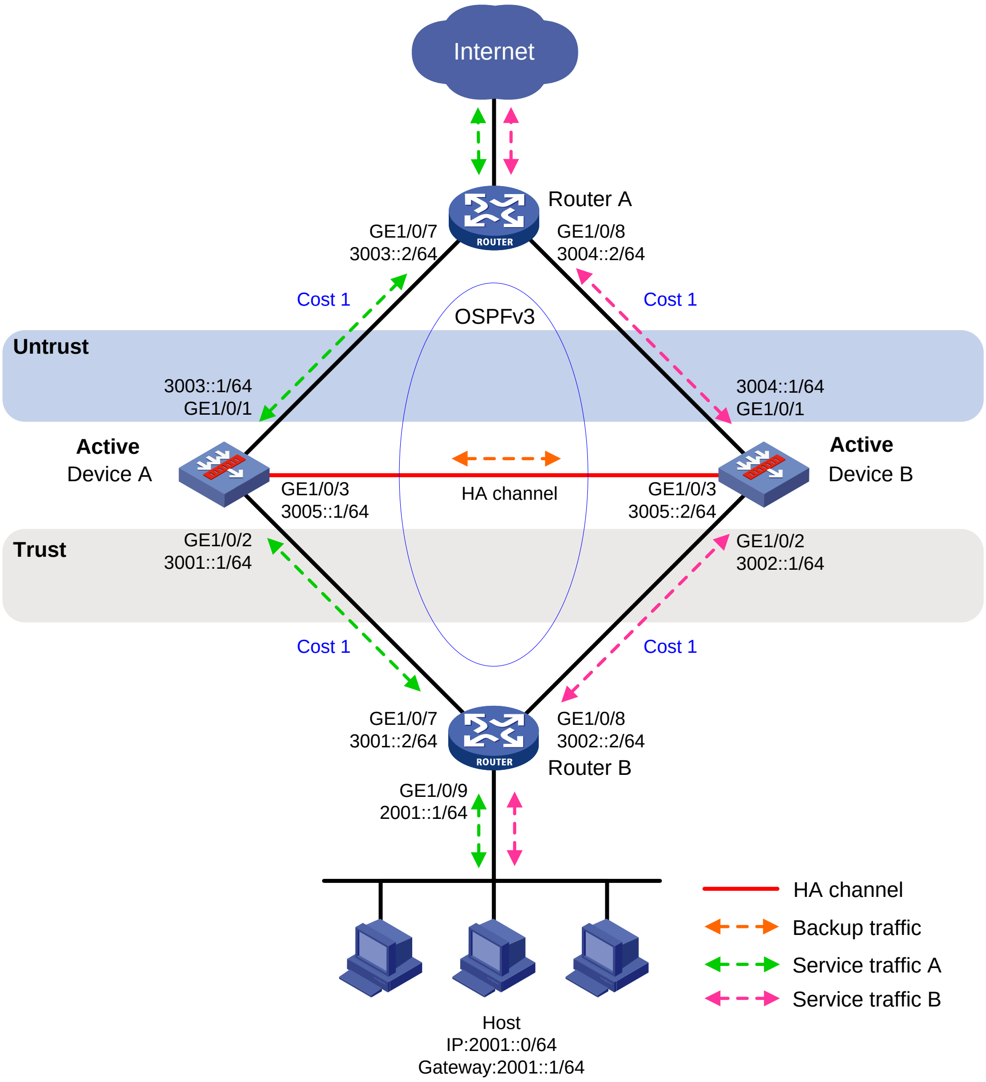
Example: Configuring dual-active hot backup in collaboration with a routing protocol

Network configuration

As shown in Figure 22, set up a hot backup system at the border between the Internet and the internal network of an enterprise to ensure service continuity.

- Configure the hot backup system to collaborate with OSPFv3.
- Configure the hot backup system to operate in dual-active mode.
- Configure Device A and Device B to load share traffic.

Figure 30 Network diagram



Procedure

1. Verify that Device A and Device B have software and hardware environment consistency.
2. Configure Router A:

NOTE:

This step only provides the brief configuration procedure.

# Assign 3003::2/64 to GigabitEthernet 1/0/7.

# Assign 3004::2/64 to GigabitEthernet 1/0/8.

# Configure OSPFv3 for Device A, Device B, and the routers to have Layer 3 reachability.

# Configure per-flow load sharing for IP forwarding.

3. Configure Router B:

NOTE:

This step only provides the brief configuration procedure.

# Assign 3001::2/64 to GigabitEthernet 1/0/7.

# Assign 3002::2/64 to GigabitEthernet 1/0/8.

# Configure OSPFv3 for Device A, Device B, and the routers to have Layer 3 reachability.

# Configure per-flow load sharing for IP forwarding.

4. Configure Device A:
   a. Assign an IP address to GigabitEthernet 1/0/1.
      ```
      <DeviceA> system-view
      [DeviceA] interface gigabitethernet 1/0/1
      [DeviceA-GigabitEthernet1/0/1] ipv6 address 3003::1/64
      [DeviceA-GigabitEthernet1/0/1] ipv6 address auto link-local
      [DeviceA-GigabitEthernet1/0/1] quit
      ```
      # Assign IP addresses to other interfaces in the same way. (Details not shown.)
   b. Add interfaces to security zones.
      ```
      [DeviceA] security-zone name untrust
      [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
      [DeviceA-security-zone-Untrust] quit
      [DeviceA] security-zone name trust
      [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
      [DeviceA-security-zone-Trust] quit
      ```
   c. Configure OSPFv3. Use the default OSPFv3 link cost configuration.
      ```
      [DeviceA] ospfv3 1
      [DeviceA-ospfv3-1] router-id 2.1.1.1
      [DeviceA-ospfv3-1] quit
      [DeviceA] interface gigabitethernet 1/0/1
      [DeviceA-GigabitEthernet1/0/1] ospfv3 1 area 0
      [DeviceA-GigabitEthernet1/0/1] quit
      [DeviceA] interface GigabitEthernet1/0/2
      [DeviceA-GigabitEthernet1/0/2] ospfv3 1 area 0
      [DeviceA-GigabitEthernet1/0/2] quit
      ```
   d. Configure a security policy.

      Perform this task only on the primary device. After the hot backup system is set up, the secondary device automatically synchronizes its security policy configuration with the primary device.

      # Configure security policy rule named **trust-untrust** to permit the packets from 2001::0/64 to the Internet.
      ```
      [DeviceA] security-policy ipv6
      [DeviceA-security-policy-ipv6] rule name trust-untrust
      [DeviceA-security-policy-ipv6-3-trust-untrust] source-zone trust
      [DeviceA-security-policy-ipv6-3-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ipv6-3-trust-untrust] source-ip-subnet 2001::0 64
      [DeviceA-security-policy-ipv6-3-trust-untrust] action pass
      [DeviceA-security-policy-ipv6-3-trust-untrust] quit
      ```
      # Configure rules to permit OSPFv3 protocol packets.
      ```
      [DeviceA-security-policy-ipv6] rule name ospf1
      [DeviceA-security-policy-ipv6-4-ospf1] source-zone trust
      ```

```
[DeviceA-security-policy-ipv6-4-ospf1] destination-zone local
[DeviceA-security-policy-ipv6-4-ospf1] service ospf
[DeviceA-security-policy-ipv6-4-ospf1] action pass
[DeviceA-security-policy-ipv6-4-ospf1] quit
[DeviceA-security-policy-ipv6] rule name ospf2
[DeviceA-security-policy-ipv6-5-ospf2] source-zone local
[DeviceA-security-policy-ipv6-5-ospf2] destination-zone trust
[DeviceA-security-policy-ipv6-5-ospf2] service ospf
[DeviceA-security-policy-ipv6-5-ospf2] action pass
[DeviceA-security-policy-ipv6-5-ospf2] quit
[DeviceA-security-policy-ipv6] rule name ospf3
[DeviceA-security-policy-ipv6-6-ospf3] source-zone untrust
[DeviceA-security-policy-ipv6-6-ospf3] destination-zone local
[DeviceA-security-policy-ipv6-6-ospf3] service ospf
[DeviceA-security-policy-ipv6-6-ospf3] action pass
[DeviceA-security-policy-ipv6-6-ospf3] quit
[DeviceA-security-policy-ipv6] rule name ospf4
[DeviceA-security-policy-ipv6-7-ospf4] source-zone local
[DeviceA-security-policy-ipv6-7-ospf4] destination-zone untrust
[DeviceA-security-policy-ipv6-7-ospf4] service ospf
[DeviceA-security-policy-ipv6-7-ospf4] action pass
[DeviceA-security-policy-ipv6-7-ospf4] quit
[DeviceA-security-policy-ipv6] quit
```

e. Configure hot backup settings.

# Associate track entries with interfaces.

```
[DeviceA] track 1 interface gigabitethernet 1/0/1
[DeviceA-track-1] quit
[DeviceA] track 2 interface gigabitethernet 1/0/2
[DeviceA-track-2] quit
```

# Set up a hot backup system.

```
[DeviceA] remote-backup group
[DeviceA-remote-backup-group] remote-ipv6 3005::2
[DeviceA-remote-backup-group] local-ipv6 3005::1
[DeviceA-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceA-remote-backup-group] device-role primary
RBM_P[DeviceA-remote-backup-group] backup-mode dual-active
RBM_P[DeviceA-remote-backup-group] hot-backup enable
RBM_P[DeviceA-remote-backup-group] configuration auto-sync enable
RBM_P[DeviceA-remote-backup-group] configuration sync-check interval 12
RBM_P[DeviceA-remote-backup-group] delay-time 1
```

# Configure the hot backup system to change the link costs advertised in OSPFv3 routes to 1.

```
RBM_P[DeviceA-remote-backup-group] adjust-cost ospfv3 enable absolute 1
```

# Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_P[DeviceA-remote-backup-group] track 1
RBM_P[DeviceA-remote-backup-group] track 2
RBM_P[DeviceA-remote-backup-group] quit
```

f. Configure security services on Device A. (Details not shown.)

5. Configure Device B:

   a. Assign IP addresses to interfaces.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 3004::1/64
[DeviceB-GigabitEthernet1/0/1] ipv6 address auto link-local
[DeviceB-GigabitEthernet1/0/1] quit
```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

   b. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

   c. Configure OSPFv3. Use the default OSPFv3 link cost configuration.

```
[DeviceB] ospfv3 1
[DeviceB-ospfv3-1] router-id 3.1.1.1
[DeviceB-ospfv3-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ospfv3 1 area 0
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface GigabitEthernet1/0/2
[DeviceB-GigabitEthernet1/0/2] ospfv3 1 area 0
[DeviceB-GigabitEthernet1/0/2] quit
```

   d. Configure hot backup settings.

   # Associate track entries with interfaces.

```
[DeviceB] track 1 interface gigabitethernet 1/0/1
[DeviceB-track-1] quit
[DeviceB] track 2 interface gigabitethernet 1/0/2
[DeviceB-track-2] quit
```

   # Set up a hot backup system.

```
[DeviceB] remote-backup group
[DeviceB-remote-backup-group] remote-ipv6 3005::1
[DeviceB-remote-backup-group] local-ipv6 3005::2
[DeviceB-remote-backup-group] data-channel interface gigabitethernet 1/0/3
[DeviceB-remote-backup-group] device-role secondary
RBM_S[DeviceB-remote-backup-group] backup-mode dual-active
RBM_S[DeviceB-remote-backup-group] hot-backup enable
RBM_S[DeviceB-remote-backup-group] configuration auto-sync enable
RBM_S[DeviceB-remote-backup-group] configuration sync-check interval 12
RBM_S[DeviceB-remote-backup-group] delay-time 1
```

   # Configure the hot backup system to change the link costs advertised in OSPFv3 routes to 1.

```
RBM_S[DeviceB-remote-backup-group] adjust-cost ospfv3 enable absolute 1
```

   # Configure the hot backup system to monitor the status of track entry 1 and track entry 2.

```
RBM_S[DeviceB-remote-backup-group] track 1
RBM_S[DeviceB-remote-backup-group] track 2
```

```
RBM_S[DeviceB-remote-backup-group] quit
```

6. On the hosts, specify 2002::1 as the default gateway. (Details not shown.)

Verifying the configuration

1. Verify the configuration on Device A:

   # Verify that the hot backup channels have been set up.

```
RBM_P[DeviceA] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Primary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IPv6: 3005::1
  Remote IPv6: 3005::2    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

   # Verify that the OSPFv3 routes advertised by Device A and Device B include the same link cost.

```
RBM_P[DeviceA] display ospfv3 interface

               OSPFv3 Process 1 with Router ID 2.1.1.1

 Area: 0.0.0.0
-------------------------------------------------------------------------
 ID      State    Cost  Pri DR              BDR             Ins Name
 2       DR       1     1   2.1.1.1         1.1.1.1         0   GE1/0/1
 3       BDR      1     1   4.1.1.1         2.1.1.1         0   GE1/0/2
```

2. Verify the configuration on Device B:

   # Verify that the hot backup channels have been set up.

```
RBM_S[DeviceB] display remote-backup-group status
Remote backup group information:
  Backup mode: Dual-active
  Device management role: Secondary
  Device running status: Active
  Data channel interface: GigabitEthernet1/0/3
  Local IPv6: 3005::2
  Remote IPv6: 3005::1    Destination port: 60064
  Control channel status: Connected
  Keepalive interval: 1s
  Keepalive count: 10
  Configuration consistency check interval: 12 hour
  Configuration consistency check result: Not Performed
  Configuration backup status: Auto sync enabled
  Session backup status: Hot backup enabled
  Delay-time: 1 min
  Uptime since last switchover: 0 days, 3 hours, 11 minutes
  Switchover records:
    Time                     Status change        Cause
    2021-06-22 13:33:33      Initial to Active    Interface status changed
```

# Verify that the OSPFv3 routes advertised by Device A and Device B include the same link cost.

```
RBM_S[DeviceB] display ospfv3 interface

               OSPFv3 Process 1 with Router ID 3.1.1.1

 Area: 0.0.0.0
-------------------------------------------------------------------------
 ID      State    Cost  Pri DR              BDR             Ins Name
 2       DR       1     1   3.1.1.1         1.1.1.1         0   GE1/0/1
 3       BDR      1     1   4.1.1.1         3.1.1.1         0   GE1/0/2
```

NSFOCUS Firewall Series

NF Interface Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for configuring interfaces in bulk, Ethernet interfaces, loopback interfaces, null interfaces, and inloopback interfaces.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

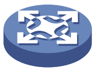
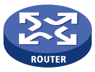
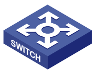
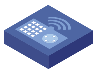
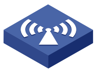
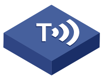
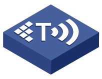
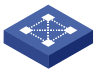
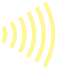
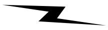
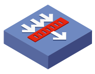
GUI conventions

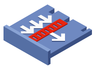
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Bulk configuring interfaces

About interface bulk configuration

You can enter interface range view to bulk configure multiple interfaces with the same feature instead of configuring them one by one. For example, you can execute the **`shutdown`** command in interface range view to shut down a range of interfaces.

To configure interfaces in bulk, you must configure an interface range and enter its view by using the **`interface range`** or **`interface range name`** command.

The interface range created by using the **`interface range`** command is not saved to the running configuration. You cannot use the interface range repeatedly. To create an interface range that can be used repeatedly, use the **`interface range name`** command.

Restrictions and guidelines: Bulk interface configuration

When you bulk configure interfaces in interface range view, follow these restrictions and guidelines:

- In interface range view, only the commands supported by the first interface in the specified interface list are available for configuration.
- Before you configure an interface as the first interface in an interface range, make sure you can enter the view of the interface by using the **`interface`** *`interface-type`* { *`interface-number`* | *`interface-number.subnumber`* } command.
- Do not assign both an aggregate interface and any of its member interfaces to an interface range. Some commands, after being executed on both an aggregate interface and its member interfaces, can break up the aggregation.
- Understand that the more interfaces you specify, the longer the command execution time.
- To guarantee bulk interface configuration performance, configure fewer than 1000 interface range names.
- After a command is executed in interface range view, one of the following situations might occur:
  - The system displays an error message and stays in interface range view. This means that the execution failed on one or multiple member interfaces.
    - If the execution failed on the first member interface, the command is not executed on any member interfaces.
    - If the execution failed on a non-first member interface, the command takes effect on the remaining member interfaces.
  - The system returns to system view. This means that:
    - The command is supported in both system view and interface view.
    - The execution failed on a member interface in interface range view and succeeded in system view.
    - The command is not executed on the subsequent member interfaces.

    You can use the **`display this`** command to verify the configuration in interface view of each member interface. In addition, if the configuration in system view is not needed, use the **`undo`** form of the command to remove the configuration.

Procedure

1. Enter system view.

   system-view

2. Create an interface range and enter interface range view.

   ○ Create an interface range without specifying a name.

     interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

   ○ Create a named interface range.

     interface range name *name* [**interface** { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>]

3. (Optional.) Display commands available for the first interface in the interface range.

   Enter a question mark (?) at the interface range prompt.

4. Use available commands to configure the interfaces.

   Available commands depend on the interface.

5. (Optional.) Verify the configuration.

   display this

Display and maintenance commands for bulk interface configuration

Execute the **display** command in any view.

Task	Command
Display information about the interface ranges created by using the **interface range name** command.	**display interface range** [**name** *name*]

Contents

Configuring Ethernet interfaces

About Ethernet interface

IMPORTANT:

Physical interfaces on firewall modules can only be used as IRF physical interfaces.

This series devices support Ethernet interfaces, Console interfaces, and USB interfaces. For the interface types and the number of interfaces supported by a device model, see the installation guide.

Configuring a management Ethernet interface

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

About this task

A management interface uses an RJ-45/LC connector. You can connect the interface to a PC for software loading and system debugging, or connect it to a remote NMS for remote system management.

Each member device in an IRF system has a management Ethernet interface. For management link backup, perform the following tasks:

1. Connect your PC to the management Ethernet interface on the master device.
2. Connect the PC to a management Ethernet interface with the same interface number on a subordinate device.

The two management Ethernet interfaces operate as follows:

- When the IRF system has multiple management Ethernet interfaces, only the management Ethernet interface on the master device processes management traffic.
- When the management Ethernet interface on the master device fails, the management Ethernet interface on the subordinate device takes over to process management traffic.
- When the management Ethernet interface on the master device recovers, it takes over to process management traffic again.

Procedure

1. Enter system view.
   system-view
2. Enter management Ethernet interface view.
   interface M-GigabitEthernet *interface-number*
3. (Optional.) Set the interface description.
   description *text*

The default setting is M-GigabitEthernet0/0/0 Interface.

4. (Optional.) Set the duplex mode for the management Ethernet interface.

   duplex { **auto** | **full** | **half** }

   By default, the duplex mode is **auto** for a management Ethernet interface.

5. (Optional.)_Set the speed for the management Ethernet interface.

   speed { **10** | **100** | **1000** | **auto** }

   By default, the speed is **auto** for a management Ethernet interface.

6. (Optional.) Shut down the interface.

   shutdown

   By default, the management Ethernet interface is up.

CAUTION:

Executing the **shutdown** command on an interface will disconnect the link of the interface and interrupt communication. Use this command with caution.

Ethernet interface naming conventions

The Ethernet interfaces are named in the format of **interface type A/B/C**. The letters that follow the interface type represent the following elements:

- **A**—IRF member ID. If the device is not in an IRF fabric, A is 1 by default.
- **B**—Card slot number. **0** indicates the interface is a fixed interface of the device. **1** indicates the interface is on expansion interface card 1. **2** indicates the interface is on expansion interface card 2.
- **C**—Port index.

Configuring common Ethernet interface settings

This section describes the settings common to Layer 2 Ethernet interfaces, Layer 3 Ethernet interfaces, and Layer 3 Ethernet subinterfaces. For more information about the settings specific to Layer 2 Ethernet interfaces, see "Configuring a Layer 2 Ethernet interface." For more information about the settings specific to Layer 3 Ethernet interfaces or subinterfaces, see "Configuring a Layer 3 Ethernet interface or subinterface."

Configuring the physical type for a combo interface (single combo interface)

About this task

A combo interface is a logical interface that physically comprises one fiber combo port and one copper combo port. The two ports share one forwarding channel and one interface view. As a result, they cannot work simultaneously. When you activate one port, the other port is automatically disabled. In the interface view, you can activate the fiber or copper combo port, and configure other port attributes such as the interface rate and duplex mode.

Hardware and feature compatibility

△ **CAUTION:**

- In the BootWare menu, fiber combo ports are not available.
- When the fiber combo port is active, it supports only the speeds autonegotiation and 1000 Mbps, and supports only duplex modes full and autonegotiation. If the copper combo port is configured with any other speed or duplex settings, the settings do not take effect after it is switched to the fiber combo port.

Models	**Feature compatibility**
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Prerequisites

Before you configure combo interfaces, complete the following tasks:

- Determine the combo interfaces on your device. Identify the two physical interfaces that belong to each combo interface according to the marks on the device panel.
- Use the **`display interface`** command to determine which port (fiber or copper) of each combo interface is active:
  - If the copper port is active, the output includes "Media type is twisted pair, Port hardware type is 1000_BASE_T."
  - If the fiber port is active, the output does not include this information.

  Also, you can use the **`display this`** command in the view of each combo interface to display the combo interface configuration:
  - If the fiber port is active, the **`combo enable fiber`** command exists in the output.
  - If the copper port is active, the **`combo enable fiber`** command does not exist in the output.

Procedure

1. Enter system view.
   `system-view`
2. Enter Ethernet interface view.
   `interface` *`interface-type interface-number`*
3. Activate the copper combo port or fiber combo port.
   `combo enable` `{` **`copper`** `|` **`fiber`** `}`
   By default, the copper combo port is active.

Configuring basic settings of an Ethernet interface

About this task

You can configure an Ethernet interface to operate in one of the following duplex modes:

- **Full-duplex mode**—The interface can send and receive packets simultaneously.
- **Half-duplex mode**—The interface can only send or receive packets at a given time.
- **Autonegotiation mode**—The interface negotiates a duplex mode with its peer.

You can set the speed of an Ethernet interface or enable it to automatically negotiate a speed with its peer.

Restrictions and guidelines

The **shutdown** and **loopback** commands are mutually exclusive.

Procedure

1. Enter system view.

   system-view

2. Enter Ethernet interface view.

   interface *interface-type interface-number*

3. Set the description for the Ethernet interface.

   description *text*

   The default setting is *interface-name* **Interface**. For example, **GigabitEthernet1/0/1 Interface**.

4. Set the duplex mode for the Ethernet interface.

   duplex { **auto** | **full** | **half** }

   By default, the duplex mode is **auto** for Ethernet interfaces.

   Fiber ports do not support the **half** keyword.

5. Set the speed for the Ethernet interface.

   speed { **10** | **100** | **1000** | **auto** }

   By default, an interface autonegotiates its speed.

6. Set the expected bandwidth for the Ethernet interface.

   bandwidth *bandwidth-value*

   By default, the expected bandwidth (in kbps) is the interface baud rate divided by 1000.

7. Bring up the Ethernet interface.

   undo shutdown

   By default, an Ethernet interface is up.

Configuring basic settings of an Ethernet subinterface

About this task

By default, a Layer 3 Ethernet subinterface processes packets for only the VLAN whose ID is the same as the subinterface number.

Restrictions and guidelines for Ethernet subinterface basic settings

- To transmit and receive packets through an Ethernet subinterface, you must associate it with a VLAN. For more information, see VLAN termination configuration in *Layer 2—LAN Switching Configuration Guide*.
- To transmit packets between a local Ethernet subinterface and a remote Ethernet subinterface, configure them with the same subinterface number and VLAN ID.
- Before creating a Layer 3 Ethernet subinterface, do not reserve a resource for the VLAN interface whose interface number is the subinterface number. After you reserve a VLAN interface resource, do not create a Layer 3 Ethernet subinterface whose subinterface number is the VLAN interface number. A Layer 3 Ethernet subinterface uses the VLAN interface resource in processing tagged packets whose VLAN ID matches the subinterface number. For more information about reserving resources for VLAN interfaces, see *Layer 2—LAN Switching Configuration Guide*.
- The **shutdown** command cannot be configured on an Ethernet interface in a loopback test.

Procedure

1. Enter system view.

   `system-view`

2. Create an Ethernet subinterface.

   `interface` *`interface-type interface-number.subnumber`*

3. Set the description for the Ethernet subinterface.

   `description` *`text`*

   The default setting is *interface-name* **Interface**. For example, **GigabitEthernet1/0/1.1 Interface**.

4. Set the expected bandwidth for the Ethernet subinterface.

   `bandwidth` *`bandwidth-value`*

   By default, the expected bandwidth (in kbps) is the interface baud rate divided by 1000.

5. Bring up the Ethernet subinterface.

   `undo shutdown`

   By default, an Ethernet subinterface is up.

Configuring the link mode of an Ethernet interface

About this task

Ethernet interfaces operate differently depending on the hardware structure of interface cards:

- Some Ethernet interfaces can operate only as Layer 2 Ethernet interfaces (in bridge mode).
- Some Ethernet interfaces can operate only as Layer 3 Ethernet interfaces (in route mode).
- Some Ethernet interfaces can operate either as Layer 2 or Layer 3 Ethernet interfaces. You can set the link mode to bridge or route for these Ethernet interfaces.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Configure the link mode of the Ethernet interface.

   `port link-mode` { **`bridge`** | **`route`** }

   By default, an Ethernet interface operates in Layer 3 mode.

> △ **CAUTION:**
>
> Changing the link mode of an Ethernet interface also restores all commands (except **`shutdown`** and **`combo enable`**) on the Ethernet interface to their defaults in the new link mode.

Configuring jumbo frame support

About this task

Jumbo frames are frames larger than a device-specific size and are typically received by an Ethernet interface during high-throughput data exchanges, such as file transfers. For more information, see *Interface Command Reference*.

The Ethernet interface processes jumbo frames in the following ways:

- When the Ethernet interface is configured to deny jumbo frames (by using the **`undo jumboframe enable`** command), the Ethernet interface discards jumbo frames.
- When the Ethernet interface is configured with jumbo frame support, the Ethernet interface performs the following operations:
    - Processes jumbo frames within the specified length.
    - Discards jumbo frames that exceed the specified length.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Configure jumbo frame support.

   `jumboframe enable` [*`size`*]

   By default, the device allows jumbo frames within the specified length to pass through. For the maximum length of jumbo frames, see *Interface Command Reference*.

   If you set the *`size`* argument multiple times, the most recent configuration takes effect.

Configuring dampening on an Ethernet interface

About this task

The interface dampening feature uses an exponential decay mechanism to prevent excessive interface flapping events from adversely affecting routing protocols and routing tables in the network. Suppressing interface state change events protects the system resources.

If an interface is not dampened, its state changes are reported. For each state change, the system also generates an SNMP trap and log message.

After a flapping interface is dampened, it does not report its state changes to the CPU. For state change events, the interface only generates SNMP trap and log messages.

Parameters

- **Penalty**—The interface has an initial penalty of 0. When the interface flaps, the penalty increases by 1000 for each down event until the ceiling is reached. It does not increase for up events. When the interface stops flapping, the penalty decreases by half each time the half-life timer expires until the penalty drops to the reuse threshold.
- **Ceiling**—The penalty stops increasing when it reaches the ceiling.
- **Suppress-limit**—The accumulated penalty that triggers the device to dampen the interface. In dampened state, the interface does not report its state changes to the CPU. For state change events, the interface only generates SNMP traps and log messages.
- **Reuse-limit**—When the accumulated penalty decreases to this reuse threshold, the interface is not dampened. Interface state changes are reported to the upper layers. For each state change, the system also generates an SNMP trap and log message.
- **Decay**—The amount of time (in seconds) after which a penalty is decreased.
- **Max-suppress-time**—The maximum amount of time the interface can be dampened. If the penalty is still higher than the reuse threshold when this timer expires, the penalty stops increasing for down events. The penalty starts to decrease until it drops below the reuse threshold.

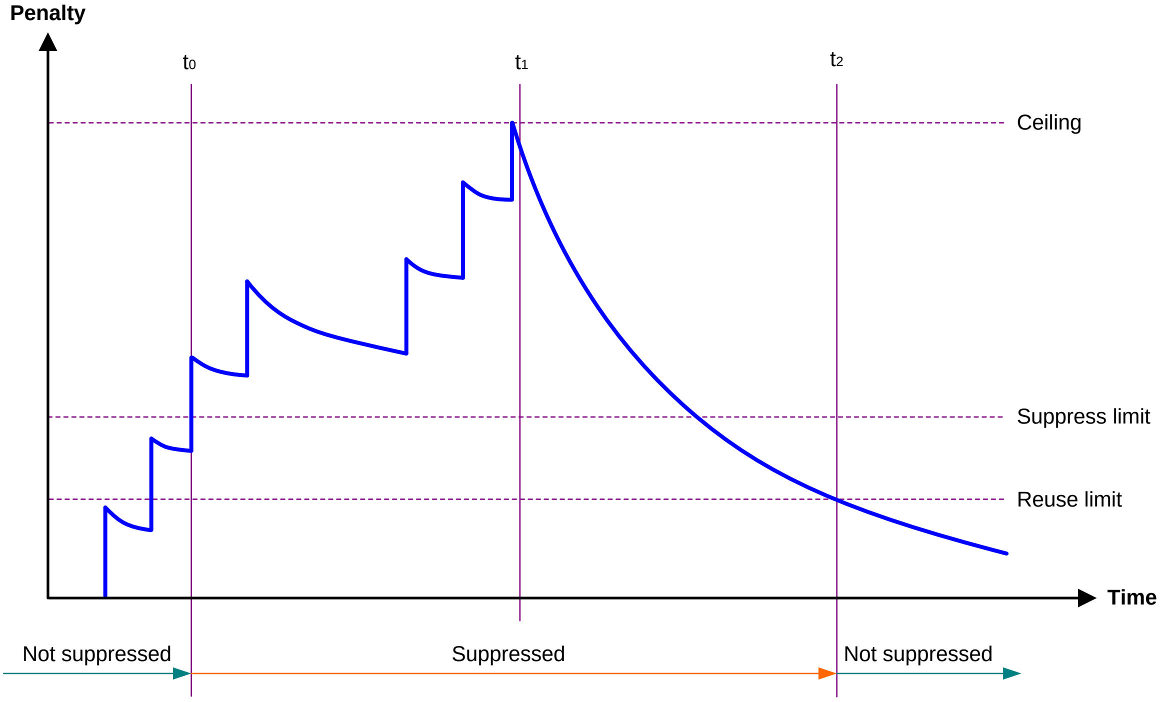
When configuring the **`dampening`** command, follow these rules to set the values mentioned above:

- The ceiling is equal to $2^{(\text{Max-suppress-time/Decay})}$ × reuse-limit. It is not user configurable.

- The configured suppress limit is lower than or equal to the ceiling.
- The ceiling is lower than or equal to the maximum suppress limit supported.

Figure 1 shows the change rule of the penalty value. The lines t_0 and t_2 indicate the start time and end time of the suppression, respectively. The period from t_0 to t_2 indicates the suppression period, t_0 to t_1 indicates the max-suppress-time, and t_1 to t_2 indicates the complete decay period.

Figure 1 Change rule of the penalty value



Restrictions and guidelines

- The **`dampening`** command does not take effect on the administratively down events. When you execute the **`shutdown`** command, the penalty restores to 0, and the interface reports the down event to the upper-layer protocols.

Do not enable this feature on an interface that has spanning tree protocols or Smart Link enabled.

Procedure

1. Enter system view.
   `system-view`
2. Enter Ethernet interface view.
   `interface` *`interface-type interface-number`*
3. Enable dampening on the interface.
   `dampening` `[` *`half-life reuse suppress max-suppress-time`* `]`
   By default, interface dampening is disabled on Ethernet interfaces.

Configuring storm suppression

About this task

The storm suppression feature ensures that the size of a particular type of traffic (broadcast, multicast, or unknown unicast traffic) does not exceed the threshold on an interface. When the broadcast, multicast, or unknown unicast traffic on the interface exceeds this threshold, the system discards packets until the traffic drops below this threshold.

Both storm suppression and storm control can suppress storms on an interface. Storm suppression uses the chip to suppress traffic. Storm suppression has less impact on the device performance than storm control, which uses software to suppress traffic.

Restrictions and guidelines

- For the traffic suppression result to be determined, do not configure storm control together with storm suppression for the same type of traffic. For more information about storm control, see "Configuring storm control on an Ethernet interface."
- The configured suppression threshold value in pps or kbps might be converted into a multiple of the step value supported by the chip. As a result, the effective suppression threshold might be different from the configured one. For information about the suppression threshold that takes effect, see the prompt on the device.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Enable broadcast suppression and set the broadcast suppression threshold.

   `broadcast-suppression` { *`ratio`* | **`pps`** *`max-pps`* | **`kbps`** *`max-kbps`* }

   By default, broadcast suppression is disabled.

4. Enable multicast suppression and set the multicast suppression threshold.

   `multicast-suppression` { *`ratio`* | **`pps`** *`max-pps`* | **`kbps`** *`max-kbps`* }

   By default, multicast suppression is disabled.

5. Enable unknown unicast suppression and set the unknown unicast suppression threshold.

   `unicast-suppression` { *`ratio`* | **`pps`** *`max-pps`* | **`kbps`** *`max-kbps`* }

   By default, unknown unicast suppression is disabled.

Configuring generic flow control on an Ethernet interface

About this task

To avoid dropping packets on a link, you can enable generic flow control at both ends of the link. When traffic congestion occurs at the receiving end, the receiving end sends a flow control (Pause) frame to ask the sending end to suspend sending packets. Generic flow control includes the following types:

- **TxRx-mode generic flow control**—Enabled by using the **`flow-control`** command. With TxRx-mode generic flow control enabled, an interface can both send and receive flow control frames:
  - When congestion occurs, the interface sends a flow control frame to its peer.
  - When the interface receives a flow control frame from its peer, it suspends sending packets to its peer.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Enable generic flow control.

```
flow-control
```

By default, generic flow control is disabled on an Ethernet interface.

By default, generic flow control is disabled on an Ethernet interface.

Setting the statistics polling interval

About this task

By setting the statistics polling interval, you can collect statistics of packets and analyze packets at the specified interval. Based on the interface traffic statistics, you can take traffic control measures promptly to avoid network congestion and service interruption.

- When network congestion is detected, you can set the statistics polling interval to be smaller than 300 seconds (30 seconds when congestion deteriorates). Then, check traffic distribution on interfaces within a short period of time. For data packets that cause congestion, take traffic control measures.
- When the network bandwidth is sufficient and services are operating normally, you can set the statistics polling interval to be greater than 300 seconds. Once traffic parameter anomalies occur, modify the statistics polling interval promptly so that you can observe the traffic parameter trend in real time.

To display the interface statistics collected in the last statistics polling interval, use the **display interface** command. To clear the interface statistics, use the **reset counters interface** command.

A device supports either the system view settings or the Ethernet interface view settings.

- The statistics polling interval configured in system view takes effect on all Ethernet interface.
- The statistics polling interval configured in Ethernet interface view takes effect only on the current interface.
- For an interface, the configuration in its Ethernet interface view takes priority. The configuration in system view is used when the configuration in Ethernet interface view is the default.

Restrictions and guidelines for setting the statistics polling interval

- This feature is not applicable to interfaces assigned to contexts in shared mode.
- As a best practice, use the default setting when you set the statistics polling interval in system view. A short statistics polling interval might decrease the system performance and result in inaccurate statistics.

Setting the statistics polling interval in system view

1. Enter system view.

```
system-view
```

2. Set the statistics polling interval.

```
flow-interval interval
```

By default, the statistics polling interval is 300 seconds.

Setting the statistics polling interval in Ethernet interface view

1. Enter system view.

```
system-view
```

2. Enter Ethernet interface view.

```
interface interface-type interface-number
```

3. Set the statistics polling interval for the Ethernet interface.

```
flow-interval interval
```

By default, the statistics polling interval is 300 seconds.

Enabling subinterface rate statistics collection on an Ethernet interface

Restrictions and guidelines

This feature is resource intensive. When you use this feature, make sure you fully understand its impact on system performance.

After you enable subinterface rate statistics collection on an Ethernet interface, the device periodically refreshes the rate statistics on the subinterfaces of this Ethernet interface.

Procedure

1. Enter system view.
   `system-view`
2. Enter Ethernet interface view.
   `interface` *`interface-type interface-number`*
3. Enable subinterface rate statistics collection on the Ethernet interface.
   `sub-interface rate-statistic`
   By default, subinterface rate statistics collection is disabled on an Ethernet interface.
4. (Optional.) View the subinterface rate statistics.
   `display interface`

Enabling loopback testing on an Ethernet interface

About this task

Perform this task to determine whether an Ethernet link works correctly.

Loopback testing includes the following types:

- **Internal loopback testing**—Tests the device where the Ethernet interface resides. The Ethernet interface sends outgoing packets back to the local device. If the device fails to receive the packets, the device fails.
- **External loopback testing**—Tests the inter-device link. The Ethernet interface sends incoming packets back to the remote device. If the remote device fails to receive the packets, the inter-device link fails.

Restrictions and guidelines

- After you enable this feature on an Ethernet interface, the interface does not forward data traffic.
- The **`shutdown`** and **`loopback`** commands are mutually exclusive.
- After you enable this feature on an Ethernet interface, the Ethernet interface switches to full duplex mode. After you disable this feature, the Ethernet interface restores to its duplex setting.

Procedure

1. Enter system view.
   `system-view`
2. Enter Ethernet interface view.
   `interface` *`interface-type interface-number`*
3. Enable loopback testing.

```
loopback { external | internal }
```

Restoring the default settings for an interface

Restrictions and guidelines

△ **CAUTION:**

This feature might interrupt ongoing network services. Make sure you are fully aware of the impacts of this feature when you use it in a live network.

This feature might fail to restore the default settings for some commands because of command dependencies or system restrictions. You can use the **display this** command in interface view to check for these commands and perform their **undo** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message to resolve the problem.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter Ethernet interface view or Ethernet subinterface view.

   ```
   interface interface-type { interface-number |
   interface-number.subnumber }
   ```

3. Restore the default settings for the interface.

   ```
   default
   ```

Configuring a Layer 2 Ethernet interface

Setting the MDIX mode of an Ethernet interface

IMPORTANT:

Fiber ports do not support the MDIX mode setting.

About this task

A physical Ethernet interface has eight pins, each of which plays a dedicated role. For example, pins 1 and 2 transmit signals, and pins 3 and 6 receive signals. You can use both crossover and straight-through Ethernet cables to connect copper Ethernet interfaces. To accommodate these types of cables, a copper Ethernet interface can operate in one of the following Medium Dependent Interface-Crossover (MDIX) modes:

- **MDIX mode**—Pins 1 and 2 are receive pins and pins 3 and 6 are transmit pins.
- **MDI mode**—Pins 1 and 2 are transmit pins and pins 3 and 6 are receive pins.
- **AutoMDIX mode**—The interface negotiates pin roles with its peer.

NOTE:

This feature does not take effect on pins 4, 5, 7, and 8 of physical Ethernet interfaces.

- Pins 4, 5, 7, and 8 of interfaces operating at 10 Mbps or 100 Mbps do not receive or transmit signals.
- Pins 4, 5, 7, and 8 of interfaces operating at 1000 Mbps or higher rates receive and transmit signals.

Restrictions and guidelines

To enable a copper Ethernet interface to communicate with its peer, set the MDIX mode of the interface by following these guidelines:

- Typically, set the MDIX mode of the interface to AutoMDIX. Set the MDIX mode of the interface to MDI or MDIX only when the device cannot determine the cable type.
- When a straight-through cable is used, configure the interface to operate in an MDIX mode different than its peer.
- When a crossover cable is used, perform one of the following tasks:
    - Configure the interface to operate in the same MDIX mode as its peer.
    - Configure either end to operate in AutoMDIX mode.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Set the MDIX mode of the Ethernet interface.

   `mdix-mode` `{` **`automdix`** `|` **`mdi`** `|` **`mdix`** `}`

   By default, a copper Ethernet interface operates in auto mode to negotiate pin roles with its peer.

Configuring storm control on an Ethernet interface

About this task

Storm control compares multicast and broadcast traffic regularly with their respective traffic thresholds on an Ethernet interface. For each type of traffic, storm control provides a lower threshold and an upper threshold.

Depending on your configuration, when a particular type of traffic exceeds its upper threshold, the interface performs either of the following operations:

- **Blocks this type of traffic and forwards other types of traffic**—Even though the interface does not forward the blocked type of traffic, it still counts the type of traffic. When the blocked type of traffic drops below the lower threshold, the interface begins to forward the traffic.
- **Goes down automatically**—The interface goes down automatically and stops forwarding any traffic. When the type of traffic exceeding the upper threshold drops below the lower threshold, the interface does not automatically come up. To bring up the interface, use the **`undo shutdown`** command or disable the storm control feature.

You can configure an Ethernet interface to output threshold event traps and log messages when monitored traffic meets one of the following conditions:

- Exceeds the upper threshold.
- Drops below the lower threshold.

Both storm suppression and storm control can suppress storms on an interface. Storm suppression uses the chip to suppress traffic. Storm suppression has less impact on the device performance than storm control, which uses software to suppress traffic. For more information about storm suppression, see "Configuring storm suppression."

Storm control uses a complete polling cycle to collect traffic data, and analyzes the data in the next cycle. An interface takes one to two polling intervals to take a storm control action.

Restrictions and guidelines

For the traffic suppression result to be determined, do not configure storm control together with storm suppression for the same type of traffic.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the statistics polling interval of the storm control module.

   storm-constrain interval *interval*

   The default setting is 10 seconds.

   For network stability, use the default or set a longer statistics polling interval.

3. Enter Ethernet interface view.

   interface *interface-type interface-number*

4. Enable storm control, and set the lower and upper thresholds for broadcast, multicast, or unknown unicast traffic.

   storm-constrain { **broadcast** | **multicast** } **pps** *upperlimit lowerlimit*

   By default, storm control is disabled.

5. Set the control action to take when monitored traffic exceeds the upper threshold.

   storm-constrain control { **block** | **shutdown** }

   By default, storm control is disabled.

6. Enable the Ethernet interface to output log messages when it detects storm control threshold events.

   storm-constrain enable log

   By default, the Ethernet interface outputs log messages when monitored traffic exceeds the upper threshold or drops below the lower threshold from a value above the upper threshold.

7. Enable the Ethernet interface to send storm control threshold event traps.

   storm-constrain enable trap

   By default, the Ethernet interface sends traps when monitored traffic exceeds the upper threshold or drops below the lower threshold from the upper threshold from a value above the upper threshold.

Configuring an interface to operate in promiscuous mode

About this task

By default, a Layer 3 Ethernet interface does not operate in promiscuous mode. In this case, the interface accepts and processes only packets destined to the MAC address of the interface, and drops packets destined to any other MAC address.

For an interface to snoop all packets received, use this command to configure the interface to operate in promiscuous mode. In this mode, a Layer 3 Ethernet interface does not perform MAC address filtering, and accepts and processes all packets regardless whether they are destined to the MAC address of the interface.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 Ethernet interface view.

   interface *interface-type interface-number*

3. Configure the interface to operate in promiscuous mode.

port promiscuous-mode

By default, an interface does not operate in promiscuous mode.

Configuring a Layer 3 Ethernet interface or subinterface

Setting the MTU for an Ethernet interface or subinterface

About this task

The maximum transmission unit (MTU) determines the maximum number of bytes in a single IP packet that can be sent. The length of an IP packet refers to the number of bytes starting from the IP header to the payload.

When the IP layer receives an IP data packet to be sent, the IP layer determines the local destination interface of the packet and obtains the MTU of the interface. The IP layer compares the MTU with the length of the data packet to be sent. If the length is greater than the MTU, the IP layer fragments the packet. The length of a fragment can be smaller than or equal to the MTU to ensure that big packets are not lost on the network.

As a best practice, use the default MTU. When the packet length or the packet receiver changes, you can adjust the MTU as needed. When configuring the MTU, follow these restrictions and guidelines:

- If the configured MTU is small but the packet length is large, the following events might occur:
  - Packets will be dropped when they are forwarded by hardware.
  - Packets will be fragmented into too many fragments when packets are forwarded through the CPUs, which affects normal data transmission.
- If the configured MTU is too large, the MTU will exceed the receiving capabilities of the receiver or a device along the transmission path. As a result, packets will be fragmented or even dropped, which increases the network transmission load and affects data transmission.

Restrictions and guidelines

The MTU of an Ethernet interface affects the fragmentation and reassembly of IP packets on the interface. Typically, you do not need to modify the MTU of an interface.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type* { *interface-number* | *interface-number.subnumber* }

3. Set the interface MTU.

   mtu *size*

   The default setting is 1500 bytes.

Setting the MAC address of an Ethernet interface

About this task

In a network, when the Layer 3 Ethernet interfaces of different devices have the same MAC address, the devices might fail to communicate correctly. To eliminate the MAC address conflicts, use the **mac-address** command to modify the MAC addresses of Layer 3 Ethernet interfaces.

Additionally, a Layer 3 Ethernet subinterface uses the MAC address of its main interface by default. The MAC address of a Layer 3 Ethernet subinterface must be the same as that of its main interface. To modify the MAC address of a Layer 3 Ethernet subinterface, first use the **`mac-address`** command to modify the MAC address of the main interface, and then modify the MAC address of the Layer 3 Ethernet subinterface.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Set the interface MAC address.

   `mac-address` *`mac-address`*

   By default, no MAC address is set for a Layer 3 Ethernet interface.

Enabling destination MAC filtering

About this task

Typically, use the default settings.

With this feature enabled, when an interface receives a packet, the interface operates as follows:

- If the destination MAC address of the packet is the MAC address of the interface, the interface accepts and processes the packet.
- If the destination MAC address of the packet is not the MAC address of the interface, the interface drops the packet.

With this feature disabled, an interface accepts and processes a packet, without checking the destination MAC address of the packet.

Restrictions and guidelines

This feature takes effect only on Layer 3 Ethernet interfaces/subinterfaces, Layer 3 aggregate interfaces, and Layer 3 Reth interfaces. These interfaces are referred to as interfaces in this feature.

Procedure

1. Enter system view.

   `system-view`

2. Enable destination MAC filtering.

   `mac-address-filter enable`

   By default, destination MAC filtering is enabled.

Enabling statistics collection on a Layer 3 Ethernet subinterface

About this task

This feature is resource intensive. When you use this feature, make sure you fully understand its impact on system performance.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 Ethernet subinterface view.
   interface *interface-type interface-number.subnumber*
3. Enable statistics collection on the Layer 3 Ethernet subinterface.
   traffic-statistic enable

   By default, statistics collection is disabled on a Layer 3 Ethernet subinterface.
4. (Optional.) Display the subinterface traffic statistics.
   display interface

   display counters

   The **Input** and **Output** fields in the **display interface** command output display the subinterface traffic statistics.

Display and maintenance commands

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display interface traffic statistics.	**display counters** { **inbound** | **outbound** } **interface** [*interface-type* [*interface-number* | *interface-number.subnumber*]]
Display traffic rate statistics of interfaces in up state over the last statistics polling interval.	**display counters rate** { **inbound** | **outbound** } **interface** [*interface-type* [*interface-number* | *interface-number.subnumber*]]
Display the Ethernet module statistics.	**display ethernet statistics slot** *slot-number*
Display the operational and status information of the specified interfaces.	**display interface** [*interface-type* [*interface-number* | *interface-number.subnumber*]] [**brief** [**description** | **down**]]
Display the status and packet statistics of interfaces.	**display interface link-info** [**main**]
Display operating status and information of all interfaces except subinterfaces.	**display interface** [*interface-type*] [**brief** [**description** | **down**]] **main**
Display information about dropped packets on the specified interfaces.	**display packet-drop** { **interface** [*interface-type* [*interface-number*]] | **summary** }
Display information about storm control on the specified interfaces.	**display storm-constrain** [**broadcast** | **multicast**] [**interface** *interface-type interface-number*]
Clear interface statistics.	**reset counters interface** [*interface-type* [*interface-number* | *interface-number.subnumber*]]
Clear the Ethernet module statistics.	**reset ethernet statistics** [**slot** *slot-number*]
Clear the statistics of dropped packets	**reset packet-drop interface**

Task	Command
on the specified interfaces.	[*interface-type* [*interface-number*]]

Contents

Configuring loopback, null, and inloopback interfaces

This chapter describes how to configure a loopback interface, a null interface, and an inloopback interface.

About loopback, null, and inloopback interfaces

About loopback interfaces

A loopback interface is a virtual interface. The physical layer state of a loopback interface is always up unless the loopback interface is manually shut down. Because of this benefit, loopback interfaces are widely used in the following scenarios:

- **Configuring a loopback interface address as the source address of the IP packets that the device generates**—Because loopback interface addresses are stable unicast addresses, they are usually used as device identifications.

  When you configure a rule on an authentication or security server, you can configure it to permit or deny packets carrying the loopback interface address of a device. This simplifies your configuration and achieves the effect of permitting or denying packets that the device generates. To use a loopback interface address as the source address of IP packets, make sure the loopback interface is reachable from the peer by performing routing configuration. All data packets sent to the loopback interface are considered packets sent to the device itself, so the device does not forward these packets.
- **Using a loopback interface in dynamic routing protocols**—With no router ID configured for a dynamic routing protocol, the system selects the highest loopback interface IP address as the router ID. In BGP, to avoid interruption of BGP sessions due to physical port failure, you can use a loopback interface as the source interface of BGP packets.

About null interfaces

A null interface is a virtual interface and is always up, but you cannot use it to forward data packets or configure it with an IP address or link layer protocol. The null interface provides a simpler way to filter packets than ACL. You can filter undesired traffic by transmitting it to a null interface instead of applying an ACL. For example, if you specify a null interface as the next hop of a static route to a network segment, any packets routed to the network segment are dropped.

About inloopback interfaces

An inloopback interface is a virtual interface created by the system, which cannot be configured or deleted. The physical layer and link layer protocol states of an inloopback interface are always up. All IP packets sent to an inloopback interface are considered packets sent to the device itself and are not forwarded.

Configuring a loopback interface

1. Enter system view.

   ```
   system-view
   ```
2. Create a loopback interface and enter loopback interface view.

```
interface loopback interface-number
```

3. Configure the interface description.

```
description text
```

The default setting is *interface name* **Interface** (for example, **LoopBack1 Interface**).

4. Configure the expected bandwidth of the loopback interface.

```
bandwidth bandwidth-value
```

By default, the expected bandwidth of a loopback interface is 0 kbps.

5. Bring up the loopback interface.

```
undo shutdown
```

By default, a loopback interface is up.

Configuring a null interface

1. Enter system view.

```
system-view
```

2. Enter null interface view.

```
interface null 0
```

Interface Null 0 is the default null interface on the device and cannot be manually created or removed.

Only one null interface, Null 0, is supported on the device. The null interface number is always 0.

3. Configure the interface description.

```
description text
```

The default setting is NULL0 Interface.

Restoring the default settings for an interface

Restrictions and guidelines

△ **CAUTION:**

This feature might interrupt ongoing network services. Make sure you are fully aware of the impact of this feature when you use it on a live network.

This feature might fail to restore the default settings for some commands because of command dependencies or system restrictions. You can use the **display this** command in interface view to check for these commands and perform their **undo** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message to resolve the problem.

Procedure

1. Enter system view.

```
system-view
```

2. Enter loopback interface view or null interface view.

   ○ `interface loopback interface-number`

   ○ `interface null 0`

3. Restore the default settings for the interface.

```
default
```

Display and maintenance commands for loopback, null, and inloopback interfaces

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about the inloopback interface.	**display interface** [**inloopback** [**0**]] [**brief** [**description** | **down**]]
Display information about the specified or all loopback interfaces.	**display interface** [**loopback** [*interface-number*]] [**brief** [**description** | **down**]]
Display information about the null interface.	**display interface** [**null** [**0**]] [**brief** [**description** | **down**]]
Clear the statistics on the specified or all loopback interfaces.	**reset counters interface** [**loopback** [*interface-number*]]
Clear the statistics on the null interface.	**reset counters interface** [**null** [**0**]]

NSFOCUS Firewall Series

NF Layer 2—LAN Switching

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for LAN switching features, including MAC address table, Ethernet link aggregation, VLANs, VLAN termination, spanning tree, LLDP, and Layer 2 forwarding.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

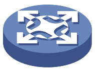
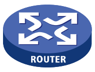
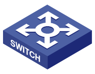
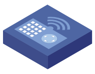
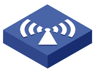
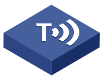
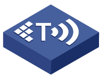
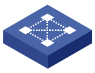
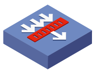
GUI conventions

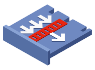
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring the MAC address table

About the MAC address table

An Ethernet device uses a MAC address table to forward frames. A MAC address entry includes a destination MAC address, an outgoing interface, and a VLAN ID. When the device receives a frame, it uses the destination MAC address of the frame to look for a match in the MAC address table.

- The device forwards the frame out of the outgoing interface in the matching entry if a match is found.
- The device floods the frame in the VLAN of the frame if no match is found.

How a MAC address entry is created

The entries in the MAC address table include entries automatically learned by the device and entries manually added.

MAC address learning

The device can automatically populate its MAC address table by learning the source MAC addresses of incoming frames on each interface.

The device performs the following operations to learn the source MAC address of incoming packets:

1. Checks the source MAC address (for example, MAC-SOURCE) of the frame.
2. Looks up the source MAC address in the MAC address table.
   - The device updates the entry if an entry is found.
   - The device adds an entry for MAC-SOURCE and the incoming port if no entry is found.

When the device receives a frame destined for MAC-SOURCE after learning this source MAC address, the device performs the following operations:

1. Finds the MAC-SOURCE entry in the MAC address table.
2. Forwards the frame out of the port in the entry.

The device performs the learning process for each incoming frame with an unknown source MAC address until the table is fully populated.

Manually configuring MAC address entries

Dynamic MAC address learning does not distinguish between illegitimate and legitimate frames, which can invite security hazards. When Host A is connected to Port A, a MAC address entry will be learned for the MAC address of Host A (for example, MAC A). When an illegal user sends frames with MAC A as the source MAC address to Port B, the device performs the following operations:

1. Learns a new MAC address entry with Port B as the outgoing interface and overwrites the old entry for MAC A.
2. Forwards frames destined for MAC A out of Port B to the illegal user.

As a result, the illegal user obtains the data of Host A. To improve the security for Host A, manually configure a static entry to bind Host A to Port A. Then, the frames destined for Host A are always sent out of Port A. Other hosts using the forged MAC address of Host A cannot obtain the frames destined for Host A.

Types of MAC address entries

A MAC address table can contain the following types of entries:

- **Static entries**—A static entry is manually added to forward frames with a specific destination MAC address out of the associated interface, and it never ages out. A static entry has higher priority than a dynamically learned one.
- **Dynamic entries**—A dynamic entry can be manually configured or dynamically learned to forward frames with a specific destination MAC address out of the associated interface. A dynamic entry might age out. A manually configured dynamic entry has the same priority as a dynamically learned one.
- **Blackhole entries**—A blackhole entry is manually configured and never ages out. A blackhole entry is configured for filtering out frames with a specific source or destination MAC address. For example, to block all frames destined for or sourced from a user, you can configure the MAC address of the user as a blackhole MAC address entry. A blackhole entry has higher priority than a dynamically learned one.

A static or blackhole MAC address entry can overwrite a dynamic MAC address entry, but not vice versa. A static entry and a blackhole entry cannot overwrite each other.

MAC address table tasks at a glance

All MAC address table configuration tasks are optional.

To configure the MAC address table, perform the following tasks:

- Configuring MAC address entries
  - Adding or modifying a static or dynamic MAC address entry
  - Adding or modifying a blackhole MAC address entry
- Setting the aging timer for dynamic MAC address entries
- Configuring MAC address learning
  - Disabling MAC address learning
  - Setting the MAC learning limit
  - Configuring the unknown frame forwarding rule after the MAC learning limit is reached
  - Assigning MAC learning priority to interfaces
- Enabling SNMP notifications for the MAC address table

Configuring MAC address entries

About MAC address entry-based frame forwarding

A frame whose source MAC address matches different types of MAC address entries is processed differently.

Type	Description
Static MAC address entry	Forwards the frame according to the destination MAC address regardless of whether the frame's ingress interface is the same as that in the entry.
Blackhole MAC address entry	Drops the frame.
Dynamic MAC address entry	• Learns the MAC address of the frames received on a different interface from that in the entry and overwrites the original entry. • Forwards the frame received on the same interface as that in the entry and updates the aging timer for the entry.

Restrictions and guidelines for MAC address entry configuration

You cannot add a dynamic MAC address entry if a learned entry already exists with a different outgoing interface for the MAC address.

The manually configured static and blackhole MAC address entries cannot survive a reboot if you do not save the configuration. The manually configured dynamic MAC address entries are lost upon reboot whether or not you save the configuration.

Do not configure the following MAC addresses as static, dynamic, or blackhole MAC addresses:

- Reserved MAC addresses of the device.
- MAC addresses of Layer 3 Ethernet interfaces or subinterfaces.
- MAC addresses of Layer 3 aggregate interfaces or subinterfaces.

For information about bridge MAC addresses, see IRF in *Virtual Technologies Configuration Guide*.

Prerequisites for MAC address entry configuration

Before manually configuring a MAC address entry for an interface, make sure the VLAN in the entry has been created.

Adding or modifying a static or dynamic MAC address entry

Adding or modifying a static or dynamic MAC address entry globally

1. Enter system view.

   system-view

2. Add or modify a static or dynamic MAC address entry.

   mac-address { **dynamic** | **static** } *mac-address* **interface** *interface-type interface-number* **vlan** *vlan-id*

   By default, no MAC address entry is configured globally.

   Make sure you have assigned the interface to the VLAN.

Adding or modifying a static or dynamic MAC address entry on an interface

1. Enter system view.

   system-view

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     interface *interface-type interface-number*
   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*

3. Add or modify a static or dynamic MAC address entry.

   mac-address { **dynamic** | **static** } *mac-address* **vlan** *vlan-id*

   By default, no MAC address entry is configured on an interface.

   Make sure you have assigned the interface to the VLAN.

Adding or modifying a blackhole MAC address entry

1. Enter system view.

   system-view

2. Add or modify a blackhole MAC address entry.

   mac-address blackhole *mac-address* **vlan** *vlan-id*

   By default, no blackhole MAC address entry is configured.

Setting the aging timer for dynamic MAC address entries

About this task

For security and efficient use of table space, the MAC address table uses an aging timer for each dynamic MAC address entry. If a dynamic MAC address entry is not updated before the aging timer expires, the device deletes the entry. This aging mechanism ensures that the MAC address table can promptly update to accommodate latest network topology changes.

A stable network requires a longer aging interval, and an unstable network requires a shorter aging interval.

An aging interval that is too long might cause the MAC address table to retain outdated entries. As a result, the MAC address table resources might be exhausted, and the MAC address table might fail to update its entries to accommodate the latest network changes.

An interval that is too short might result in removal of valid entries, which would cause unnecessary floods and possibly affect the device performance.

To reduce floods on a stable network, set a long aging timer or disable the timer to prevent dynamic entries from unnecessarily aging out. Reducing floods improves the network performance. Reducing flooding also improves the security because it reduces the chances for a data frame to reach unintended destinations.

Procedure

1. Enter system view.

   system-view

2. Set the aging timer for dynamic MAC address entries.

   mac-address timer { **aging** *seconds* | **no-aging** }

   The default setting is 300 seconds.

Disabling MAC address learning

About disabling MAC address learning

MAC address learning is enabled by default. To prevent the MAC address table from being saturated when the device is experiencing attacks, disable MAC address learning. For example, you can disable MAC address learning to prevent the device from being attacked by a large amount of frames with different source MAC addresses.

After MAC address learning is disabled, existing dynamic MAC address entries can age out.

Disabling global MAC address learning

Restrictions and guidelines

After you disable global MAC address learning, the device cannot learn MAC addresses on any interfaces.

Procedure

1. Enter system view.
   system-view
2. Disable global MAC address learning.
   undo mac-address mac-learning enable
   By default, global MAC address learning is enabled.

Disabling MAC address learning on an interface

About this task

When global MAC address learning is enabled, you can disable MAC address learning on a single interface.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.
     interface *interface-type interface-number*
   - Enter Layer 2 aggregate interface view.
     interface bridge-aggregation *interface-number*
3. Disable MAC address learning on the interface.
   undo mac-address mac-learning enable
   By default, MAC address learning is enabled on an interface.

Setting the MAC learning limit

Setting the MAC learning limit on interfaces

About this task

This feature limits the MAC address table size. A large MAC address table will degrade forwarding performance.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.
     interface *interface-type interface-number*
   - Enter Layer 2 aggregate interface view.

```
interface bridge-aggregation interface-number
```

3. Set the MAC learning limit on the interface.

   ```
   mac-address max-mac-count count
   ```

   By default, the maximum number of MAC addresses that an interface can learn is restricted by hardware capability of the device.

Configuring the unknown frame forwarding rule after the MAC learning limit is reached

In this document, unknown frames refer to frames whose source MAC addresses are not in the MAC address table.

About unknown frame forwarding rule configuration

You can enable or disable forwarding of unknown frames after the MAC learning limit is reached.

Configuring the device to forward unknown frames after the MAC learning limit on an interface is reached

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     ```
     interface interface-type interface-number
     ```

   - Enter Layer 2 aggregate interface view.

     ```
     interface bridge-aggregation interface-number
     ```

3. Configure the device to forward unknown frames received on the interface after the MAC learning limit on the interface is reached.

   ```
   mac-address max-mac-count enable-forwarding
   ```

   By default, the device can forward unknown frames received on an interface after the MAC learning limit on the interface is reached.

Assigning MAC learning priority to interfaces

About this task

The MAC learning priority mechanism assigns either low priority or high priority to an interface. An interface with high priority can learn MAC addresses as usual. However, an interface with low priority is not allowed to learn MAC addresses already learned on a high-priority interface.

The MAC learning priority mechanism can help defend your network against MAC address spoofing attacks. In a network that performs MAC-based forwarding, an upper layer device MAC address might be learned by a downlink interface because of a loop or attack to the downlink interface. To avoid this issue, perform the following tasks:

- Assign high MAC learning priority to an uplink interface.
- Assign low MAC learning priority to a downlink interface.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   - Enter Layer 2 Ethernet interface view.

     interface *interface-type interface-number*

   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*

3. Assign MAC learning priority to the interface.

   mac-address mac-learning priority { **high** | **low** }

   By default, low MAC learning priority is used.

Enabling SNMP notifications for the MAC address table

About this task

To report critical MAC address move events to an NMS, enable SNMP notifications for the MAC address table. For MAC address move event notifications to be sent correctly, you must also configure SNMP on the device.

When SNMP notifications are disabled for the MAC address table, the device sends the generated logs to the information center. To display the logs, configure the log destination and output rule configuration in the information center.

For more information about SNMP and information center configuration, see the network management and monitoring configuration guide for the device.

Procedure

1. Enter system view.

   system-view

2. Enable SNMP notifications for the MAC address table.

   snmp-agent trap enable mac-address

   By default, SNMP notifications are enabled for the MAC address table.

   When SNMP notifications are disabled for the MAC address table, syslog messages are sent to notify important events on the MAC address table module.

Display and maintenance commands for MAC address table

Execute **display** commands in any view.

Task	Command
Display MAC address table information.	**display mac-address** [*mac-address* [**vlan** *vlan-id*] | [[**dynamic** | **static**] [**interface** *interface-type interface-number*] | **blackhole**] [**vlan** *vlan-id*] [**count**]]

Task	Command
Display the aging timer for dynamic MAC address entries.	**display mac-address aging-time**
Display the system or interface MAC address learning state.	**display mac-address mac-learning** [**interface** *interface-type interface-number*]

Contents

Configuring Ethernet link aggregation

About Ethernet link aggregation

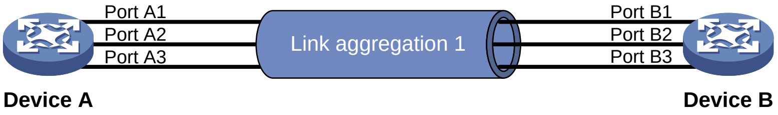
Ethernet link aggregation bundles multiple physical Ethernet links into one logical link (called an aggregate link). Link aggregation provides the following benefits:

- Increased bandwidth beyond the limits of a single individual link. In an aggregate link, traffic is distributed across the member ports.
- Improved link reliability. The member ports dynamically back up one another. When a member port fails, its traffic is automatically switched to other member ports.

Ethernet link aggregation application scenario

As shown in Figure 1, Device A and Device B are connected by three physical Ethernet links. These physical Ethernet links are combined into an aggregate link called link aggregation 1. The bandwidth of this aggregate link can reach up to the total bandwidth of the three physical Ethernet links. At the same time, the three Ethernet links back up one another. When a physical Ethernet link fails, the traffic transmitted on the failed link is switched to the other two links.

Figure 1 Ethernet link aggregation diagram



Aggregate interface, aggregation group, and member port

Each link aggregation is represented by a logical aggregate interface. Each aggregate interface has an automatically created aggregation group, which contains member ports to be used for aggregation. The type and number of an aggregation group are the same as its aggregate interface.

Supported aggregate interface types

An aggregate interface can be one of the following types:

- **Layer 2**—A Layer 2 aggregate interface is created manually. The member ports in a Layer 2 aggregation group can only be Layer 2 Ethernet interfaces.
- **Layer 3**—A Layer 3 aggregate interface is created manually. The member ports in its Layer 3 aggregation group can only be Layer 3 Ethernet interfaces.

  On a Layer 3 aggregate interface, you can create subinterfaces. A Layer 3 aggregate subinterface processes traffic only for the VLAN numbered with the same ID as the subinterface number.

The port rate of an aggregate interface equals the total rate of its Selected member ports. Its duplex mode is the same as that of the Selected member ports. For more information about Selected member ports, see "Aggregation states of member ports in an aggregation group."

Aggregation states of member ports in an aggregation group

A member port in an aggregation group can be in any of the following aggregation states:

- **Selected**—A Selected port can forward traffic.
- **Unselected**—An Unselected port cannot forward traffic.

- **Individual**—An Individual port can forward traffic as a normal physical port. This state is peculiar to the member ports of edge aggregate interfaces. A Selected or Unselected member port of an edge aggregate interface is placed in Individual state if the following events occur in sequence:
  - **a.** The member port goes down and then comes up.
  - **b.** The LACP timeout timer expires because it has not received LACPDUs.

  For more information about edge aggregate interfaces, see "Edge aggregate interface."

Operational key

When aggregating ports, the system automatically assigns each port an operational key based on port information, such as port rate and duplex mode. Any change to this information triggers a recalculation of the operational key.

In an aggregation group, all Selected ports have the same operational key.

Configuration types

Port configuration includes the attribute configuration and protocol configuration. Attribute configuration affects the aggregation state of the port but the protocol configuration does not.

Attribute configuration

To become a Selected port, a member port must have the same attribute configuration as the aggregate interface. Table 1 describes the attribute configuration.

Table 1 Attribute configuration

Feature	Attribute configuration
VLAN	VLAN attribute settings: • Permitted VLAN IDs. • PVID. • Link type (trunk, hybrid, or access). For information about VLANs, see "Configuring VLANs."

Protocol configuration

Protocol configuration of a member port does not affect the aggregation state of the member port. MAC address learning and spanning tree settings are examples of the protocol configuration.

Link aggregation modes

An aggregation group operates in one of the following modes:

- **Static**—Static aggregation is stable. An aggregation group in static mode is called a static aggregation group. The aggregation states of the member ports in a static aggregation group are not affected by the peer ports.
- **Dynamic**—An aggregation group in dynamic mode is called a dynamic aggregation group. Dynamic aggregation is implemented through IEEE 802.3ad Link Aggregation Control Protocol (LACP). The local system and the peer system automatically maintain the aggregation states of the member ports. Dynamic link aggregation reduces the administrators' workload.

How static link aggregation works

Reference port selection process

When setting the aggregation states of the ports in an aggregation group, the system automatically chooses a member port as the reference port. A Selected port must have the same operational key and attribute configurations as the reference port.

The system chooses a reference port from the member ports in up state.

The candidate reference ports are organized into different priority levels following these rules:

1. In descending order of port priority.
2. Full duplex.
3. In descending order of speed.
4. Half duplex.
5. In descending order of speed.

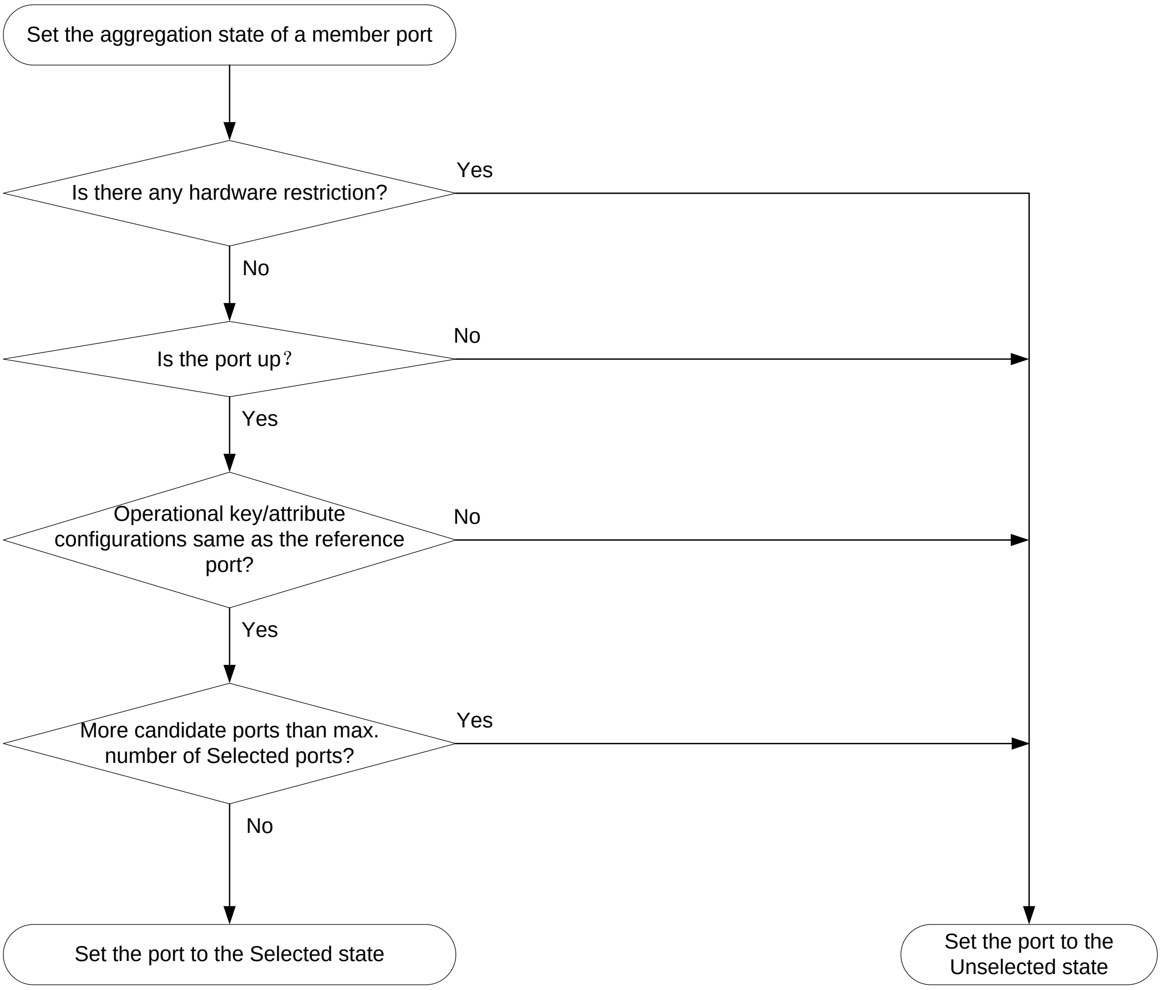
From the candidate ports with the same attribute configurations as the aggregate interface, the one with the highest priority level is chosen as the reference port.

- If multiple ports have the same priority level, the port that has been Selected (if any) is chosen. If multiple ports with the same priority level have been Selected, the one with the smallest port number is chosen.
- If multiple ports have the same priority level and none of them has been Selected, the port with the smallest port number is chosen.

Setting the aggregation state of each member port

After the reference port is chosen, the system sets the aggregation state of each member port in the static aggregation group.

Figure 2 Setting the aggregation state of a member port in a static aggregation group


Set the aggregation state of a member port
Is there any hardware restriction?
Yes
No
Is the port up？
No
Yes
Operational key/attribute configurations same as the reference port?
No
Yes
More candidate ports than max. number of Selected ports?
Yes
No
Set the port to the Selected state
Set the port to the Unselected state

After the limit on Selected ports is reached, the aggregation state of a new member port varies by following conditions:

- The port is placed in Unselected state if the port and the Selected ports have the same port priority. This mechanism prevents traffic interruption on the existing Selected ports. A device reboot can cause the device to recalculate the aggregation states of member ports.
- The port is placed in Selected state when the following conditions are met:
    - The port and the Selected ports have different port priorities, and the port has a higher port priority than a minimum of one Selected port.
    - The port has the same attribute configurations as the aggregate interface.

Any operational key or attribute configuration change might affect the aggregation states of link aggregation member ports.

Dynamic link aggregation

About LACP

Dynamic aggregation is implemented through IEEE 802.3ad Link Aggregation Control Protocol (LACP).

LACP uses LACPDUs to exchange aggregation information between LACP-enabled devices. Each member port in a dynamic aggregation group can exchange information with its peer. When a member port receives an LACPDU, it compares the received information with information received on the other member ports. In this way, the two systems reach an agreement on which ports are placed in Selected state.

LACP functions

LACP offers basic LACP functions and extended LACP functions, as described in Table 2.

Table 2 LACP functions

Category	Description
Basic LACP functions	Implemented through the basic LACPDU fields, including the system LACP priority, system MAC address, port priority, port number, and operational key.
Extended LACP functions	Implemented by extending the LACPDU with new TLV fields. Extended LACP can implement LACP MAD for the IRF feature. • If a device supports both extended LACP and IRF, it can participate in LACP MAD as either an IRF member device or an intermediate device. • If a device supports extended LACP but not IRF, it can participate in LACP MAD only as an intermediate device. For more information about IRF and the LACP MAD mechanism, see *Virtual Technologies Configuration Guide*.

LACP operating modes

LACP can operate in active or passive mode.

When LACP is operating in passive mode on a local member port and its peer port, both ports cannot send LACPDUs. When LACP is operating in active mode on either end of a link, both ports can send LACPDUs.

LACP priorities

LACP priorities include system LACP priority and port priority, as described in Table 3. The smaller the priority value, the higher the priority.

Table 3 LACP priorities

Type	Description
System LACP priority	Used by two peer devices (or systems) to determine which one is superior in link aggregation. In dynamic link aggregation, the system that has higher system LACP priority sets the Selected state of member ports on its side. The system that has lower priority sets the aggregation state of local member ports the same as their respective peer ports.
Port priority	Determines the likelihood of a member port to be a Selected port on a system. A port with a higher port priority is more likely to become Selected.

LACP timeout interval

The LACP timeout interval specifies how long a member port waits to receive LACPDUs from the peer port. If a local member port has not received LACPDUs from the peer within the LACP timeout interval plus 3 seconds, the member port considers the peer as failed.

The LACP timeout interval also determines the LACPDU sending rate of the peer. LACP timeout intervals include the following types:

- **Short timeout interval**—3 seconds. If you use the short timeout interval, the peer sends one LACPDU per second.
- **Long timeout interval**—90 seconds. If you use the long timeout interval, the peer sends one LACPDU every 30 seconds.

How dynamic link aggregation works

Choosing a reference port

The system chooses a reference port from the member ports in up state. A Selected port must have the same operational key and attribute configurations as the reference port.

The local system (the actor) and the peer system (the partner) negotiate a reference port by using the following workflow:

1. The two systems determine the system with the smaller system ID.

   A system ID contains the system LACP priority and the system MAC address.

   a. The two systems compare their LACP priority values.

      The lower the LACP priority, the smaller the system ID. If the LACP priority values are the same, the two systems proceed to step b.

   b. The two systems compare their MAC addresses.

      The lower the MAC address, the smaller the system ID.

2. The system with the smaller system ID chooses the port with the smallest port ID as the reference port.

   A port ID contains a port priority and a port number. The lower the port priority, the smaller the port ID.

   a. The system chooses the port with the lowest priority value as the reference port.

      If the ports have the same priority, the system proceeds to step b.

   b. The system compares their port numbers.

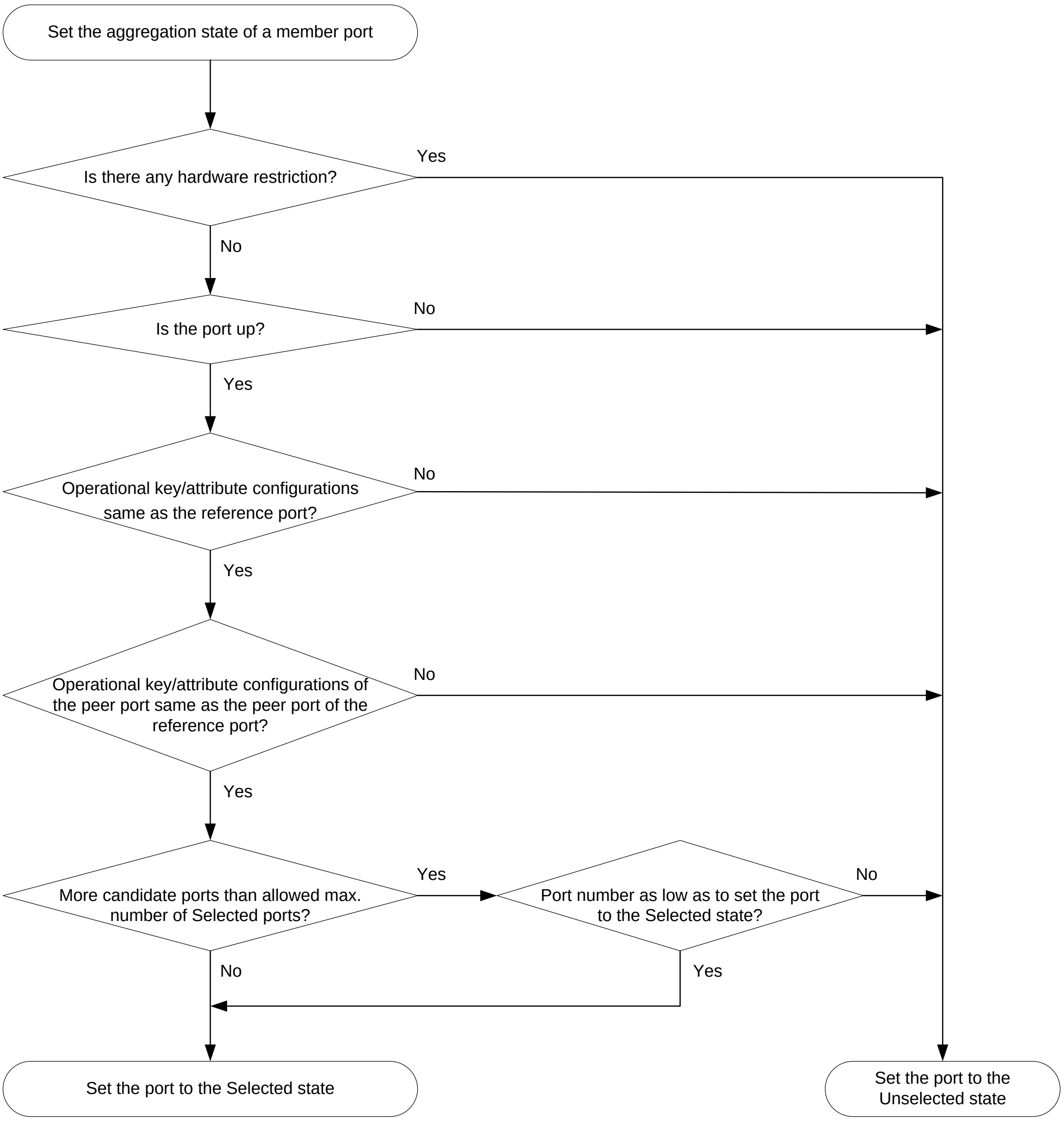
      The smaller the port number, the smaller the port ID.

      The port with the smallest port number and the same attribute configurations as the aggregate interface is chosen as the reference port.

Setting the aggregation state of each member port

After the reference port is chosen, the system with the smaller system ID sets the state of each member port on its side.

Figure 3 Setting the state of a member port in a dynamic aggregation group



Set the aggregation state of a member port
Is there any hardware restriction?
Yes
No
Is the port up?
No
Yes
Operational key/attribute configurations same as the reference port?
No
Yes
Operational key/attribute configurations of the peer port same as the peer port of the reference port?
No
Yes
More candidate ports than allowed max. number of Selected ports?
Yes
Port number as low as to set the port to the Selected state?
No
No
Yes
Set the port to the Selected state
Set the port to the Unselected state

The system with the greater system ID can detect the aggregation state changes on the peer system. The system with the greater system ID sets the aggregation state of local member ports the same as their peer ports.

When you aggregate interfaces in dynamic mode, follow these guidelines:

- A dynamic link aggregation group chooses only full-duplex ports as the Selected ports.
- For stable aggregation and service continuity, do not change the operational key or attribute configurations on any member port.
- When a member port changes to the Selected or Unselected state, its peer port changes to the same aggregation state.
- After the Selected port limit is reached, a newly joining port becomes a Selected port if it is more eligible than a current Selected port.

Edge aggregate interface

Dynamic link aggregation fails on a server-facing aggregate interface if dynamic link aggregation is configured only on the device. The device forwards traffic by using only one of the physical ports that are connected to the server.

To improve link reliability, configure the aggregate interface as an edge aggregate interface. This feature enables all member ports of the aggregation group to forward traffic. When a member port fails, its traffic is automatically switched to other member ports.

After dynamic link aggregation is configured on the server, the device can receive LACPDUs from the server. Then, link aggregation between the device and the server operates correctly.

An edge aggregate interface takes effect only when it is configured on an aggregate interface corresponding to a dynamic aggregation group.

Load sharing modes for link aggregation groups

In a link aggregation group, traffic can be load shared across the Selected ports based on any of the following modes:

- **Per-flow load sharing**—Distributes traffic on a per-flow basis. The load sharing mode classifies packets into flows and forwards packets of the same flow on the same link. This mode can be one of or a combination of the following traffic classification criteria:
    - Ingress port.
    - Source or destination IP.
    - Source or destination MAC.
    - Source or destination port number.
- **Bandwidth usage-based load sharing**—Distributes a data flow to the Selected port that had the lowest bandwidth usage when the first packet of that data flow arrived. In this mode, each flow is identified by an IP five-tuple (source and destination IP addresses, source and destination ports, and protocol). For packets that do not contain the IP five-tuple, the default load sharing mode applies.
- **Packet type-based load sharing**—Automatically selects a load sharing mode depending on the packet type. For example, the load sharing mode differs between IPv4 packets and Layer 2 packets. This mode is also called the flexible mode.

Restrictions and guidelines: Mixed use of manual and automatic link aggregation configuration

To avoid unexpected aggregation issues, do not use manual assignment and automatic link aggregation together. If you use these features together, an automatically assigned member port might move between aggregation groups or undesirably change from Selected to Unselected in some situations.

Ethernet link aggregation tasks at a glance

To configure Ethernet link aggregation, perform the following tasks:

1. Configuring link aggregations
    - Configuring a manual link aggregation
2. (Optional.) Configuring an aggregate interface
    - Setting the minimum and maximum numbers of Selected ports for an aggregation group

    - Configuring the description of an aggregate interface
    - Configuring jumbo frame support
    - Disabling the default action of selecting a Selected port for dynamic aggregation groups that have not received LACPDUs
    - Specifying ignored VLANs for a Layer 2 aggregate interface

      To have the system ignore the permit state and tagging mode of a VLAN when it decides Selected ports, perform this task.
    - Setting the MTU of a Layer 3 aggregate interface
    - Setting the expected bandwidth for an aggregate interface
    - Configuring an edge aggregate interface

      An edge aggregate interface uses all member ports to forward traffic when the aggregation peer is not enabled with dynamic link aggregation.
    - Configuring physical state change suppression on an aggregate interface
    - Shutting down an aggregate interface
    - Restoring the default settings for an aggregate interface
3. (Optional.) Configuring load sharing for link aggregation groups
    - Setting load sharing modes for link aggregation groups
    - Enabling local-first load sharing for link aggregation
4. (Optional.) Enabling forwarding acceleration for link aggregation
5. (Optional.) Enabling link-aggregation traffic redirection

   This feature redirects traffic on an unavailable Selected port to the remaining available Selected ports of an aggregation group to avoid traffic interruption.

Configuring a manual link aggregation

Restrictions and guidelines for aggregation group configuration

Layer 2 aggregation group restrictions

You cannot assign an interface to a Layer 2 aggregation group if the redundancy group node feature or MAC authentication is configured on that interface. For more information about redundancy groups, see *Virtual Technologies Configuration Guide*. For more information about MAC authentication, see *Security Configuration Guide*.

Layer 3 aggregation group restrictions

You cannot assign an interface to a Layer 3 aggregation group if any features in Table 4 are configured on that interface.

Table 4 Features incompatible with Layer 3 aggregation member interfaces

Feature on the interface	Reference
Reth interface	Reth interfaces in *Virtual Technologies Configuration Guide*
Redundancy group node	Redundancy groups in *Virtual Technologies Configuration Guide*

Aggregation member port restrictions

Deleting an aggregate interface also deletes its aggregation group and causes all member ports to leave the aggregation group.

The following restrictions apply if you have not enabled multi-VLAN termination on an aggregate interface by using the **`link-aggregation multivlan-termination`** command:

- You cannot assign both Ethernet interfaces and Ethernet subinterfaces to the aggregation group.
- You cannot create subinterfaces on an Ethernet interface that is in the aggregation group.
- You cannot assign an Ethernet interface that has subinterfaces to the aggregation group.

You cannot create aggregate subinterfaces on an aggregate interface if its aggregation group contains Ethernet subinterfaces. You cannot assign Ethernet subinterfaces to an aggregation group if its aggregate interface has aggregate subinterfaces.

Before you assign an Ethernet subinterface to an aggregation group, perform the following tasks:

- If multi-VLAN termination is not enabled on the aggregate interface, configure VLAN termination on the Ethernet subinterface. You will be unable to modify the VLAN termination configuration after you assign the subinterface to the aggregation group. To configure VLAN termination, use the following commands:
    - **`vlan-type dot1q default`**.
    - **`vlan-type dot1q untagged`**.
    - **`vlan-type dot1q vid`**.
- To assign Ethernet subinterfaces that terminate different VLANs to the same aggregation group, enable multi-VLAN termination on the aggregate interface. If multi-VLAN termination is not enabled on the aggregate interface, you must make sure the Ethernet subinterfaces to be assigned to its aggregation group terminate the same VLAN.
- If you are assigning the Ethernet subinterface to a dynamic aggregation group, specify only one VLAN ID when you execute the **`vlan-type dot1q vid`** *`vlan-id-list`* [**`loose`**] command.

Attribute and protocol configuration restrictions

For a link aggregation, attribute configuration changes on the aggregate interface are automatically synchronized to all member ports. If an attribute setting on the aggregate interface fails to be synchronized to a Selected member port, the port might change to the Unselected state. To have the port become Selected again, you can change the attribute configurations on the aggregate interface or the member port. The configurations that have been synchronized from the aggregate interface are retained on the member ports even after the aggregate interface is deleted.

Any attribute configuration change on a member port might affect the aggregation states and running services of the member ports. The system displays a warning message every time you try to change an attribute configuration setting on a member port.

The protocol configurations for an aggregate interface take effect only on the current aggregate interface. The protocol configurations for a member port take effect only when the port leaves its aggregation group.

Configuration consistency requirements

You must configure the same aggregation mode at the two ends of an aggregate link.

- For a successful static aggregation, make sure the ports at both ends of each link are in the same aggregation state.
- For a successful dynamic aggregation, make sure the ports at both ends of a link are assigned to the correct aggregation group. The two ends can automatically negotiate the aggregation state of each member port.

Configuring a Layer 2 aggregation group

Configuring a Layer 2 static aggregation group

1. Enter system view.

   system-view

2. Create a Layer 2 aggregate interface and enter Layer 2 aggregate interface view.

   interface bridge-aggregation *interface-number*

   When you create a Layer 2 aggregate interface, the system automatically creates a Layer 2 static aggregation group numbered the same as that interface.

3. Return to system view.

   quit

4. Assign an interface to the Layer 2 aggregation group:

   a. Enter Layer 2 Ethernet interface view.

      interface *interface-type interface-number*

   b. Assign the interface to the Layer 2 aggregation group.

      port link-aggregation group *group-id*

   Repeat the substeps to assign more interfaces to the aggregation group.

5. (Optional.) Set the port priority of the interface.

   link-aggregation port-priority *priority*

   The default port priority of an interface is 32768.

Configuring a Layer 2 dynamic aggregation group

1. Enter system view.

   system-view

2. Set the system LACP priority.

   lacp system-priority *priority*

   By default, the system LACP priority is 32768.

   Changing the system LACP priority might affect the aggregation states of the ports in a dynamic aggregation group.

3. Create a Layer 2 aggregate interface and enter Layer 2 aggregate interface view.

   interface bridge-aggregation *interface-number*

   When you create a Layer 2 aggregate interface, the system automatically creates a Layer 2 static aggregation group numbered the same as that interface.

4. Configure the aggregation group to operate in dynamic mode.

   link-aggregation mode dynamic

   By default, an aggregation group operates in static mode.

5. Return to system view.

   quit

6. Assign an interface to the Layer 2 aggregation group:

   a. Enter Layer 2 Ethernet interface view.

      interface *interface-type interface-number*

   b. Assign the interface to the Layer 2 aggregation group.

      port link-aggregation group *group-id*

   Repeat these substeps to assign more Layer 2 Ethernet interfaces to the aggregation group.

7. Set the LACP operating mode for the interface.
   - Set the LACP operating mode to passive.

     lacp mode passive
   - Set the LACP operating mode to active.

     undo lacp mode

   By default, LACP is operating in active mode.
8. (Optional.) Set the port priority for the interface.

   link-aggregation port-priority *priority*

   The default setting is 32768.
9. (Optional.) Set the short LACP timeout interval (3 seconds) for the interface.

   lacp period short

   By default, the long LACP timeout interval (90 seconds) is used by the interface.

   To avoid traffic interruption during an ISSU, do not set the short LACP timeout interval before performing the ISSU. For more information about ISSU, see *Fundamentals Configuration Guide*.

Configuring a Layer 3 aggregation group

Configuring a Layer 3 static aggregation group

1. Enter system view.

   system-view
2. Create a Layer 3 aggregate interface and enter Layer 3 aggregate interface view.

   interface route-aggregation *interface-number*

   When you create a Layer 3 aggregate interface, the system automatically creates a Layer 3 static aggregation group numbered the same as that interface.
3. Return to system view.

   quit
4. Assign an interface to the Layer 3 aggregation group:
   a. Enter Layer 3 Ethernet interface view.

      interface *interface-type interface-number*

   b. Assign the interface to the Layer 3 aggregation group.

      port link-aggregation group *group-id*

   Repeat the substeps to assign more interfaces to the aggregation group.
5. (Optional.) Set the port priority of the interface.

   link-aggregation port-priority *priority*

   The default port priority of an interface is 32768.

Configuring a Layer 3 dynamic aggregation group

1. Enter system view.

   system-view
2. Set the system LACP priority.

   lacp system-priority *priority*

   By default, the system LACP priority is 32768.

   Changing the system LACP priority might affect the aggregation states of the ports in the dynamic aggregation group.

3. Create a Layer 3 aggregate interface and enter Layer 3 aggregate interface view.

   `interface route-aggregation` *`interface-number`*

   When you create a Layer 3 aggregate interface, the system automatically creates a Layer 3 static aggregation group numbered the same as that interface.

4. Configure the aggregation group to operate in dynamic mode.

   `link-aggregation mode dynamic`

   By default, an aggregation group operates in static mode.

5. Return to system view.

   `quit`

6. Assign an interface to the Layer 3 aggregation group:

   a. Enter Layer 3 Ethernet interface view.

      `interface` *`interface-type interface-number`*

   b. Assign the interface to the Layer 3 aggregation group.

      `port link-aggregation group` *`group-id`*

   Repeat these two substeps to assign more Layer 3 Ethernet interfaces to the aggregation group.

7. Set the LACP operating mode for the interface.

   - Set the LACP operating mode to passive.

     `lacp mode passive`

   - Set the LACP operating mode to active.

     `undo lacp mode`

   By default, LACP is operating in active mode.

8. (Optional.) Set the port priority of the interface.

   `link-aggregation port-priority` *`priority`*

   The default setting is 32768.

9. (Optional.) Set the short LACP timeout interval (3 seconds) for the interface.

   `lacp period short`

   By default, the long LACP timeout interval (90 seconds) is used by the interface.

   To avoid traffic interruption during an ISSU, do not set the short LACP timeout interval before performing the ISSU. For more information about ISSU, see *Fundamentals Configuration Guide*.

Configuring an aggregate interface

Most settings that can be made on Layer 2 or Layer 3 Ethernet interfaces can also be made on Layer 2 or Layer 3 aggregate interfaces.

Setting the minimum and maximum numbers of Selected ports for an aggregation group

About this task

The bandwidth of an aggregate link increases as the number of Selected member ports increases. To avoid congestion, you can set the minimum number of Selected ports required for bringing up an aggregate interface.

This minimum threshold setting affects the aggregation states of aggregation member ports and the state of the aggregate interface.

- When the number of member ports eligible to be Selected ports is smaller than the minimum threshold, the following events occur:
  - The eligible member ports are placed in Unselected state.
  - The link layer state of the aggregate interface becomes down.
- When the number of member ports eligible to be Selected ports reaches or exceeds the minimum threshold, the following events occur:
  - The eligible member ports are placed in Selected state.
  - The link layer state of the aggregate interface becomes up.

The maximum number of Selected ports allowed in an aggregation group is limited by either manual configuration or hardware limitation, whichever value is smaller.

You can implement backup between two ports by performing the following tasks:

- Assigning two ports to an aggregation group.
- Setting the maximum number of Selected ports to 1 for the aggregation group.

Then, only one Selected port is allowed in the aggregation group, and the Unselected port acts as a backup port.

Restrictions and guidelines

The minimum and maximum numbers of Selected ports must be the same between the local and peer aggregation groups.

For an aggregation group, the maximum number of Selected ports must be equal to or higher than the minimum number of Selected ports.

Procedure

1. Enter system view.
   `system-view`
2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.
     `interface bridge-aggregation` *`interface-number`*
   - Enter Layer 3 aggregate interface view.
     `interface route-aggregation` *`interface-number`*
3. Set the minimum number of Selected ports for the aggregation group.
   `link-aggregation selected-port minimum` *`min-number`*
   By default, the minimum number of Selected ports is not specified for an aggregation group.
4. Set the maximum number of Selected ports for the aggregation group.
   `link-aggregation selected-port maximum` *`max-number`*
   The maximum number of Selected ports for an aggregation group is in the range of 1 to 16.

Configuring the description of an aggregate interface

About this task

You can configure the description of an aggregate interface for administration purposes, for example, describing the purpose of the interface.

Procedure

1. Enter system view.

system-view

2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.
     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.
     interface route-aggregation *interface-number*
   - Enter Layer 3 aggregate subinterface view.
     interface route-aggregation *interface-number.subnumber* }
3. Configure the interface description.
   description *text*

   By default, the description of an interface is *interface-name* **Interface**.

Configuring jumbo frame support

About this task

An aggregate interface might receive frames larger than the maximum frame size allowed by an interface during high-throughput data exchanges, such as file transfers. These frames are called jumbo frames.

How an aggregate interface processes jumbo frames depends on whether jumbo frame support is enabled on the interface.

- If configured to deny jumbo frames, the aggregate interface discards jumbo frames.
- If enabled with jumbo frame support, the aggregate interface performs the following operations:
  - Processes jumbo frames within the allowed length.
  - Discards jumbo frames that exceed the allowed length.

Procedure

1. Enter system view.
   system-view
2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.
     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.
     interface route-aggregation *interface-number*
3. Allow jumbo frames.
   jumboframe enable [*size*]

   The default setting varies by device model. For more information, see the command reference.

   If you execute this command multiple times, the most recent configuration takes effect.

Disabling the default action of selecting a Selected port for dynamic aggregation groups that have not received LACPDUs

About this task

The default port selection action applies to dynamic aggregation groups.

This action automatically chooses the port with the lowest ID from among all up member ports as a Selected port if none of them has received LACPDUs before the LACP timeout interval expires.

After this action is disabled, a dynamic aggregation group will not have any Selected ports to forward traffic if it has not received LACPDUs before the LACP timeout interval expires.

Procedure

1. Enter system view.

   `system-view`

2. Disable the default port selection action.

   `lacp default-selected-port disable`

   By default, the default port selection action is enabled for dynamic aggregation groups.

Specifying ignored VLANs for a Layer 2 aggregate interface

About this task

By default, to become Selected, the member ports must have the same VLAN permit state and tagging mode as the corresponding Layer 2 aggregate interface. To have the system ignore the permit state and tagging mode of a VLAN when choosing Selected ports, specify the VLAN as an ignored VLAN.

Restrictions and guidelines

This feature takes effect only when the link type of a Layer 2 aggregate interface is hybrid or trunk.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 2 aggregate interface view.

   `interface bridge-aggregation` *`interface-number`*

3. Specify ignored VLANs.

   `link-aggregation ignore vlan` *`vlan-id-list`*

   By default, a Layer 2 aggregate interface does not ignore any VLANs.

Setting the MTU of a Layer 3 aggregate interface

About this task

The MTU of an interface affects IP packets fragmentation and reassembly on the interface.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 aggregate interface or subinterface view.

   `interface route-aggregation` { *`interface-number`* | *`interface-number.subnumber`* }

3. Set the MTU.

   `mtu` *`size`*

   The default setting is 1500 bytes.

Setting the expected bandwidth for an aggregate interface

About this task

Expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by performing this task.

Procedure

1. Enter system view.

   `system-view`

2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*
   - Enter Layer 3 aggregate interface view.

     `interface route-aggregation` *`interface-number`*
   - Enter Layer 3 aggregate subinterface view.

     `interface route-aggregation` *`interface-number.subnumber`* }

3. Set the expected bandwidth for the interface.

   `bandwidth` *`bandwidth-value`*

   By default, the expected bandwidth (in kbps) is the interface baud rate divided by 1000.

Configuring an edge aggregate interface

Restrictions and guidelines

This configuration takes effect only on aggregate interface in dynamic mode.

Link-aggregation traffic redirection cannot operate correctly on an edge aggregate interface. For more information about link-aggregation traffic redirection, see "Enabling link-aggregation traffic redirection."

Procedure

1. Enter system view.

   `system-view`

2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*
   - Enter Layer 3 aggregate interface view.

     `interface route-aggregation` *`interface-number`*

3. Configure the aggregate interface as an edge aggregate interface.

   `lacp edge-port`

   By default, an aggregate interface does not operate as an edge aggregate interface.

Configuring physical state change suppression on an aggregate interface

About this task

The physical link state of an aggregate interface is either up or down. Each time the physical link of an interface comes up or goes down, the system immediately reports the change to the CPU. The CPU then performs the following operations:

- Notifies the upper-layer protocol modules (such as routing and forwarding modules) of the change for guiding packet forwarding.
- Automatically generates traps and logs to inform users to take the correct actions.

To prevent frequent physical link flapping from affecting system performance, configure physical state change suppression. You can configure this feature to suppress link-down events, link-up events, or both. If an event of the specified type persists when the suppression interval expires, the system reports the event to the CPU.

Restrictions and guidelines

On an interface, you can configure different suppression intervals for link-up and link-down events. If you execute the **link-delay** command multiple times for an event type, the most recent configuration takes effect on that event type.

Procedure

1. Enter system view.

   system-view

2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.

     interface route-aggregation *interface-number*

3. Configure physical state change suppression.

   link-delay [**msec**] *delay-time* [**mode** { **up** | **updown** }]

   By default, each time the physical link of an aggregate interface goes up or comes down, the system immediately reports the change to the CPU.

   To suppress only link-down events, do not specify the **mode** keyword. To suppress only link-up events, specify the **mode up** keywords. To suppress both link-down and link-up events, specify the **mode updown** keywords.

Shutting down an aggregate interface

Restrictions and guidelines

△ **CAUTION:**

The **shutdown** command will disconnect all links established on an interface. Make sure you are fully aware of the impacts of this command when you use it on a live network.

Shutting down or bringing up an aggregate interface affects the aggregation states and link states of member ports in the corresponding aggregation group as follows:

- When an aggregate interface is shut down, all its Selected ports become Unselected and all member ports go down.

- When an aggregate interface is brought up, the aggregation states of all its member ports are recalculated.

Procedure

1. Enter system view.
   system-view
2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.
     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.
     interface route-aggregation *interface-number*
   - Enter Layer 3 aggregate subinterface view.
     interface route-aggregation *interface-number.subnumber* }
3. Shut down the interface.
   shutdown

   By default, an interface is not manually shut down.

Restoring the default settings for an aggregate interface

Restrictions and guidelines

CAUTION:

The **default** command might interrupt ongoing network services. Make sure you are fully aware of the impacts of this command when you execute it on a live network.

The **default** command might fail to restore the default settings for some commands for reasons such as command dependencies and system restrictions.

To resolve this issue:

1. Use the **display this** command in interface view to identify these commands.
2. Use their **undo** forms or follow the command reference to restore their default settings.
3. If the restoration attempt still fails, follow the error message instructions to resolve the issue.

Procedure

1. Enter system view.
   system-view
2. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.
     interface bridge-aggregation *interface-number*
   - Enter Layer 3 aggregate interface view.
     interface route-aggregation *interface-number*
   - Enter Layer 3 aggregate subinterface view.
     interface route-aggregation *interface-number.subnumber* }
3. Restore the default settings for the aggregate interface.
   default

Configuring load sharing for link aggregation groups

Setting load sharing modes for link aggregation groups

About this task

You can set the global or group-specific load sharing mode. A link aggregation group preferentially uses the group-specific load sharing mode. If the group-specific load sharing mode is not available, the group uses the global load sharing mode.

Setting the global link-aggregation load sharing mode

1. Enter system view.

   `system-view`

2. Set the global link-aggregation load sharing mode.

   `link-aggregation global load-sharing mode` { **`destination-ip`** | **`destination-mac`** | **`destination-port`** | **`ingress-port`** | **`source-ip`** | **`source-mac`** | **`source-port`** } *

   By default, the system automatically chooses a load sharing mode depending on the packet type.

Setting the group-specific load sharing mode

1. Enter system view.

   `system-view`

2. Enter aggregate interface view.

   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*

   - Enter Layer 3 aggregate interface view.

     `interface route-aggregation` *`interface-number`*

3. Set the load sharing mode for the aggregation group.

   `link-aggregation load-sharing mode` { **`destination-ip`** | **`destination-mac`** | **`destination-port`** | **`source-ip`** | **`source-mac`** | **`source-port`** } *

   By default, an aggregation group uses the global link-aggregation load sharing mode.

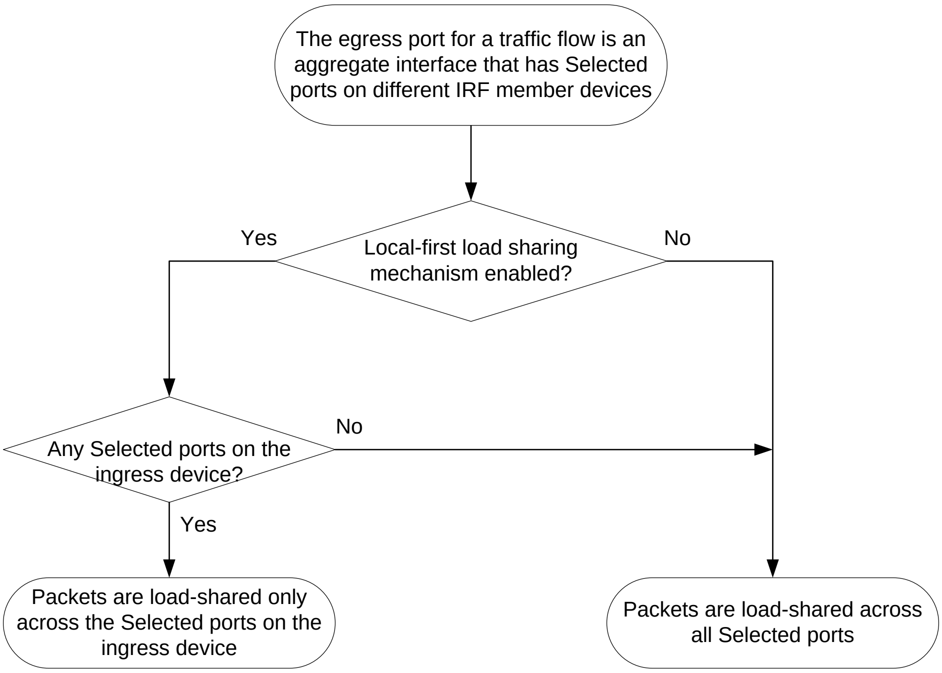
Enabling local-first load sharing for link aggregation

About this task

Use local-first load sharing in a multichassis link aggregation scenario to distribute traffic preferentially across member ports on the ingress device.

When you aggregate ports on different member devices in an IRF fabric, you can use local-first load sharing to reduce traffic on IRF links, as shown in Figure 4. For more information about IRF, see *Virtual Technologies Configuration Guide*.

Figure 4 Load sharing for multichassis link aggregation in an IRF fabric


The egress port for a traffic flow is an aggregate interface that has Selected ports on different IRF member devices
Yes
Local-first load sharing mechanism enabled?
No
Any Selected ports on the ingress device?
No
Yes
Packets are load-shared only across the Selected ports on the ingress device
Packets are load-shared across all Selected ports

Enabling local-first load sharing for link aggregation globally

1. Enter system view.

   `system-view`

2. Enable local-first load sharing for link aggregation globally.

   `link-aggregation load-sharing mode local-first`

   By default, local-first load sharing is enabled globally.

Enabling forwarding acceleration for link aggregation

About this task

This feature accelerates traffic forwarding on an aggregate interface whose member ports are located on multiple slots.

Restrictions and guidelines

Forwarding acceleration takes effect on an aggregate interface only when it is enabled both globally and on the aggregate interface.

Procedure

1. Enter system view.

   `system-view`

2. Enable forwarding acceleration globally.

   `link-aggregation global forwarding-acceleration enable`

   By default, forwarding acceleration is disabled globally.

3. Enter aggregate interface view.
   - Enter Layer 2 aggregate interface view.

```
interface bridge-aggregation interface-number
```

- Enter Layer 3 aggregate interface view.

```
interface route-aggregation interface-number
```

4. Enable forwarding acceleration on the interface.

```
link-aggregation forwarding-acceleration enable
```

By default, forwarding acceleration is enabled on aggregate interfaces.

Enabling link-aggregation traffic redirection

About link-aggregation traffic redirection

This feature operates on dynamic link aggregation groups. It redirects traffic on a Selected port to the remaining available Selected ports of an aggregation group if the port is shut down by using the **shutdown** command or the slot that hosts the port reboots.

NOTE:

The device does not redirect traffic to member ports that become Selected during the traffic redirection process.

This feature ensures zero packet loss for known unicast traffic, but does not protect unknown unicast traffic.

Restrictions and guidelines for link-aggregation traffic redirection

Link-aggregation traffic redirection applies only to dynamic link aggregation groups.

As a best practice, enable link-aggregation traffic redirection on a per-interface basis. If you enable this feature globally, communication with a third-party peer device might be affected if the peer is not compatible with this feature.

To prevent traffic interruption, enable link-aggregation traffic redirection at both ends of the aggregate link.

To prevent packet loss that might occur at a reboot, do not enable the spanning tree feature together with link-aggregation traffic redirection.

Link-aggregation traffic redirection does not operate correctly on an edge aggregate interface.

Enabling link-aggregation traffic redirection globally

1. Enter system view.

```
system-view
```

2. Enable link-aggregation traffic redirection globally.

```
link-aggregation lacp traffic-redirect-notification enable
```

By default, link-aggregation traffic redirection is disabled globally.

Display and maintenance commands for Ethernet link aggregation

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about aggregate interfaces.	**display interface** [{ **bridge-aggregation** | **route-aggregation** } [*interface-number*]] [**brief** [**description** | **down**]]
Display the local system ID.	**display lacp system-id**
Display the global or group-specific link-aggregation load sharing modes.	**display link-aggregation load-sharing mode** [**interface** [{ **bridge-aggregation** | **route-aggregation** } *interface-number*]]
Display forwarding information for the specified traffic flow.	**display link-aggregation load-sharing path interface** { **bridge-aggregation** | **route-aggregation** } *interface-number* **ingress-port** *interface-type* *interface-number* [**route**] { { **destination-ip** *ip-address* | **destination-ipv6** *ipv6-address* } | { **source-ip** *ip-address* | **source-ipv6** *ipv6-address* } | **destination-mac** *mac-address* | **destination-port** *port-id* | **ethernet-type** *type-number* | **ip-protocol** *protocol-id* | **source-mac** *mac-address* | **source-port** *port-id* | **vlan** *vlan-id* } * **slot** *slot-number*
Display detailed link aggregation information about link aggregation member ports.	**display link-aggregation member-port** [*interface-list* | **auto**]
Display summary information about all aggregation groups.	**display link-aggregation summary**
Display detailed information about the specified aggregation groups.	**display link-aggregation verbose** [{ **bridge-aggregation** | **route-aggregation** } [*interface-number*]]
Clear statistics about the specified aggregate interfaces.	**reset counters interface** [{ **bridge-aggregation** | **route-aggregation** } [*interface-number*]]
Clear LACP statistics about the specified link aggregation member ports.	**reset lacp statistics** [**interface** *interface-list*]

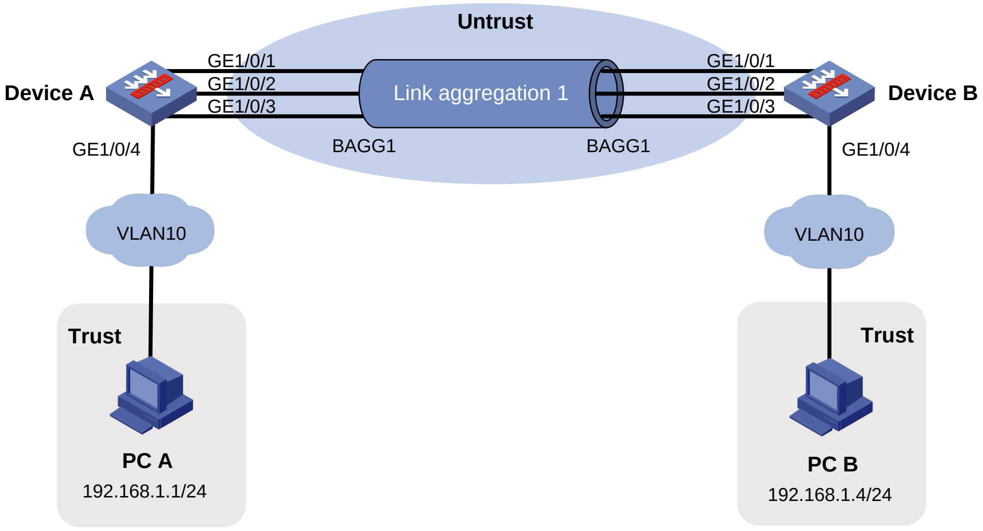
Ethernet link aggregation configuration examples

Example: Configuring a Layer 2 static aggregation group

Network configuration

As shown in Figure 5, configure a Layer 2 static aggregation group on both Device A and Device B to aggregate the links between them.

Figure 5 Network diagram



Procedure

1. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/4 to operate in Layer 2 mode.

```
<DeviceA> system-view
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/4
[DeviceA-if-range] port link-mode bridge
[DeviceA-if-range] quit
```

2. Create VLAN 10, and assign interface GigabitEthernet 1/0/4 to VLAN 10.

```
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/4
[DeviceA-vlan10] quit
```

3. Configure a Layer 2 static aggregation group:

# Configure Layer 2 aggregate interface Bridge-Aggregation 1 as a trunk port and assign it to VLANs 1 and 10.

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 1 10
[DeviceA-Bridge-Aggregation1] quit
```

# Assign interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 to link aggregation group 1.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
[DeviceA-if-range] port link-aggregation group 1
[DeviceA-if-range] quit
```

4. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 as trunk ports and assign them to VLANs 1 and 10.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
[DeviceA-if-range] port link-type trunk
[DeviceA-if-range] port trunk permit vlan 1 10
[DeviceA-if-range] quit
```

5. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/4 vlan 10
[Device-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[Device-security-zone-Untrust] import interface bridge-aggregation 1 vlan 1 10
[Device-security-zone-Untrust] quit
```

6. Configure a security policy:

Configure rules to permit traffic between the **Trust** and **Untrust** security zones, so the devices can communicate with each other:

# Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

7. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired
Aggregate Interface: Bridge-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
```

```
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1
  GE1/0/3          S       32768    1
```

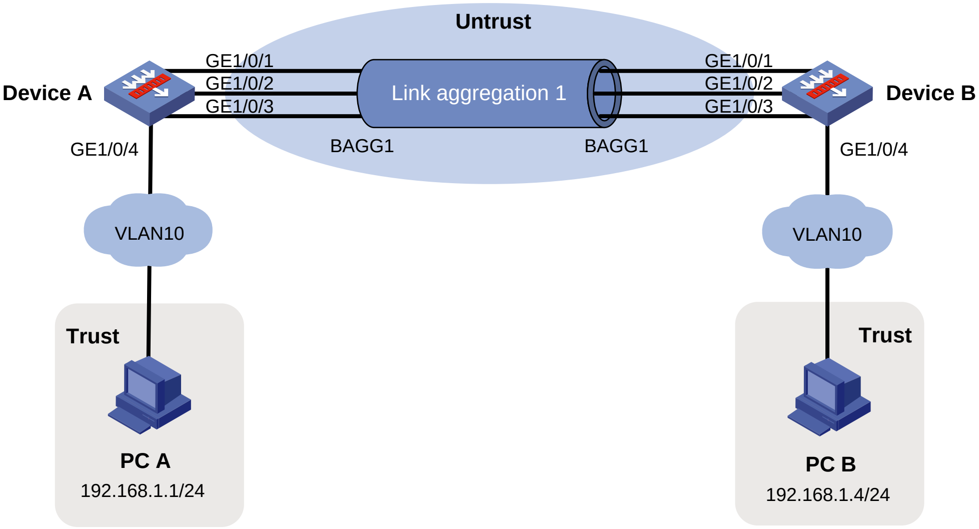
The output shows that link aggregation group 1 is a Layer 2 static aggregation group that contains three Selected ports.

Example: Configuring a Layer 2 dynamic aggregation group

Network configuration

As shown in Figure 6, configure a Layer 2 dynamic aggregation group on both Device A and Device B to aggregate the links between them.

Figure 6 Network diagram



Procedure

1. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/4 to operate in Layer 2 mode.

```
<DeviceA> system-view
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/4
[DeviceA-if-range] port link-mode bridge
[DeviceA-if-range] quit
```

2. Create VLAN 10, and assign interface GigabitEthernet 1/0/4 to VLAN 10.

```
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/4
[DeviceA-vlan10] quit
```

3. Configure a Layer 2 dynamic aggregation group:

# Configure Layer 2 aggregate interface Bridge-Aggregation 1 as a trunk port, assign it to VLANs 1 and 10, and set the link aggregation mode to dynamic.

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 1 10
[DeviceA-Bridge-Aggregation1] link-aggregation mode dynamic
```

```
[DeviceA-Bridge-Aggregation1] quit
```

# Assign interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 to link aggregation group 1.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
[DeviceA-if-range] port link-aggregation group 1
[DeviceA-if-range] quit
```

4. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 as trunk ports and assign them to VLANs 1 and 10.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
[DeviceA-if-range] port link-type trunk
[DeviceA-if-range] port trunk permit vlan 1 10
[DeviceA-if-range] quit
```

5. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/4 vlan 10
[Device-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[Device-security-zone-Untrust] import interface bridge-aggregation 1 vlan 1 10
[Device-security-zone-Untrust] quit
```

6. Configure a security policy:

Configure rules to permit traffic between the **Trust** and **Untrust** security zones, so the devices can communicate with each other:

# Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

7. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired
Aggregate Interface: Bridge-Aggregation1
Creation Mode: Manual
Aggregation Mode: Dynamic
Loadsharing Type: Shar
System ID: 0x8000, 000f-e267-6c6a
Local:
  Port             Status  Priority Oper-Key  Flag
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1         {ACDEF}
  GE1/0/2          S       32768    1         {ACDEF}
  GE1/0/3          S       32768    1         {ACDEF}
Remote:
  Actor            Partner Priority Oper-Key  SystemID               Flag
--------------------------------------------------------------------------------
  GE1/0/1          1       32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/2          2       32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/3          3       32768    1         0x8000, 000f-e267-57ad {ACDEF}
```

The output shows that link aggregation group 1 is a Layer 2 dynamic aggregation group that contains three Selected ports.

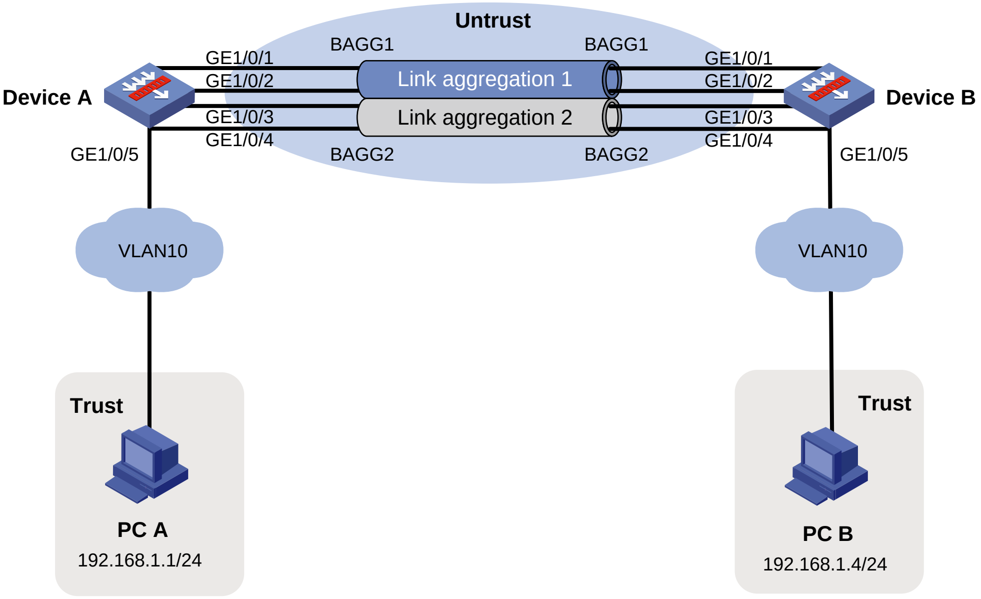
Example: Configuring Layer 2 aggregation load sharing

Network configuration

As shown in Figure 7, perform the following tasks:

- Configure Layer 2 static aggregation groups 1 and 2 on Device A and Device B, respectively.
- Configure link aggregation groups 1 and 2 to load share traffic across aggregation group member ports.
    - Configure link aggregation group 1 to load share packets based on source MAC addresses.
    - Configure link aggregation group 2 to load share packets based on destination MAC addresses.

Figure 7 Network diagram



Procedure

1. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/5 to operate in Layer 2 mode.

```
<DeviceA> system-view
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/5
[DeviceA-if-range] port link-mode bridge
[DeviceA-if-range] quit
```

2. Create VLAN 10, and assign interface GigabitEthernet 1/0/5 to VLAN 10.

```
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/5
[DeviceA-vlan10] quit
```

3. Configure Layer 2 aggregation groups:

# Configure Layer 2 aggregate interface Bridge-Aggregation 1 as a trunk port and assign it to VLANs 1 and 10, and configure it to load share packets based on source MAC addresses..

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 1 10
[DeviceA-Bridge-Aggregation1] link-aggregation load-sharing mode source-mac
[DeviceA-Bridge-Aggregation1] quit
```

# Assign interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 to link aggregation group 1.

```
[DeviceA] interface range GigabitEthernet1/0/1 to GigabitEthernet1/0/2
[DeviceA-if-range] port link-aggregation group 1
[DeviceA-if-range] quit
```

# Configure Layer 2 aggregate interface Bridge-Aggregation 2 as a trunk port and assign it to VLANs 1 and 10, and configure it to load share packets based on destination MAC addresses..

```
[DeviceA] interface bridge-aggregation 2
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 1 10
```

```
[DeviceA-Bridge-Aggregation1] link-aggregation load-sharing mode destination-mac
[DeviceA-Bridge-Aggregation1] quit
```

# Assign interfaces GigabitEthernet 1/0/3 and GigabitEthernet 1/0/4 to link aggregation group 2.

```
[DeviceA] interface range gigabitethernet 1/0/3 to gigabitethernet 1/0/4
[DeviceA-if-range] port link-aggregation group 2
[DeviceA-if-range] quit
```

4. Configure interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/4 as trunk ports and assign them to VLANs 1 and 10.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/4
[DeviceA-if-range] port link-type trunk
[DeviceA-if-range] port trunk permit vlan 1 10
[DeviceA-if-range] quit
```

5. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/5 vlan 10
[Device-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[Device-security-zone-Untrust] import interface bridge-aggregation 1 vlan 1 10
[Device-security-zone-Untrust] import interface bridge-aggregation 2 vlan 1 10
[Device-security-zone-Untrust] quit
```

6. Configure a security policy:

Configure rules to permit traffic between the **Trust** and **Untrust** security zones, so the devices can communicate with each other:

# Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

7. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
```

```
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Bridge-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1

Aggregate Interface: Bridge-Aggregation2
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/3          S       32768    2
  GE1/0/4          S       32768    2
```

The output shows that:

- Link aggregation groups 1 and 2 are both load-shared Layer 2 static aggregation groups.
- Each aggregation group contains two Selected ports.

# Display all the group-specific load sharing modes on Device A.

```
[DeviceA] display link-aggregation load-sharing mode interface
Bridge-Aggregation1 Load-Sharing Mode:
source-mac address

Bridge-Aggregation2 Load-Sharing Mode:
destination-mac address
```

The output shows that:

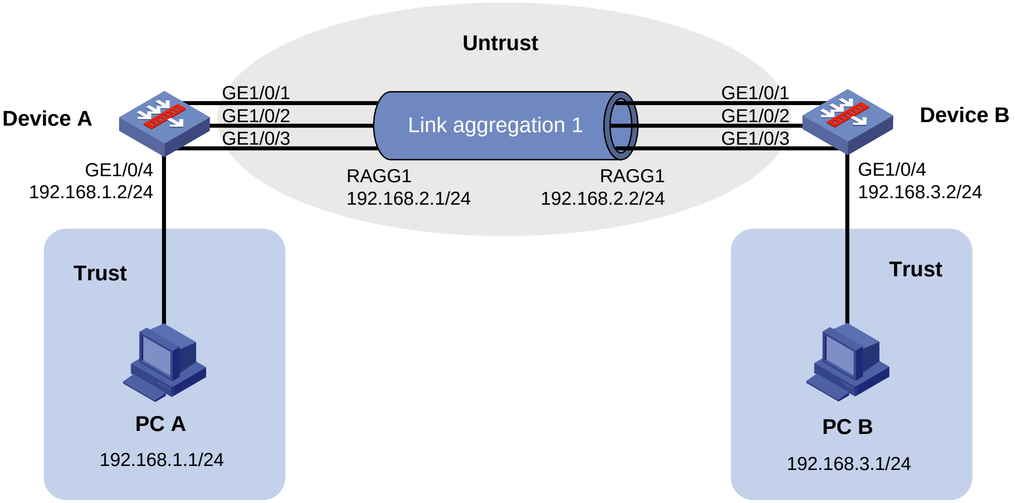
- Link aggregation group 1 distributes packets based on source MAC addresses.
- Link aggregation group 2 distributes packets based on destination MAC addresses.

Example: Configuring a Layer 3 static aggregation group

Network configuration

As shown in Figure 8, configure a Layer 3 static aggregation group on both Device A and Device B to aggregate the links between them.

Figure 8 Network diagram



Procedure

1. Configure a Layer 3 static aggregation group:

   # Create Layer 3 aggregate interface Route-Aggregation 1, and assign an IP address to the aggregate interface.

```
<DeviceA> system-view
[DeviceA] interface route-aggregation 1
[DeviceA-Route-Aggregation1] ip address 192.168.2.1 24
[DeviceA-Route-Aggregation1] quit
```

   # Assign Layer 3 Ethernet interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 to aggregation group 1.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
[DeviceA-if-range] port link-aggregation group 1
[DeviceA-if-range] quit
```

2. Assign an IP address to interface GigabitEthernet 1/0/4.

```
[DeviceA] interface GigabitEthernet 1/0/4
[DeviceA-GigabitEthernet1/0/4] ip address 192.168.1.2 24
[DeviceA-GigabitEthernet1/0/4] quit
```

3. Configure settings for routing.

   This example configure a static route for PC B, and the next hop is 192.168.2.2.

```
[DeviceA] ip route-static 192.168.3.1 24 192.168.2.2
```

4. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/4
[Device-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[Device-security-zone-Untrust] import interface route-aggregation 1
[Device-security-zone-Untrust] quit
```

5. Configure a security policy:

   Configure rules to permit traffic between PC A and PC B:

   # Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone.

```
[DeviceA] security-policy ip
```

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] source-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-0-trust-untrust] destination-ip-subnet 192.168.3.0 24
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] source-ip-subnet 192.168.3.0 24
[DeviceA-security-policy-ip-1-untrust-trust] destination-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

6. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1
  GE1/0/3          S       32768    1
```

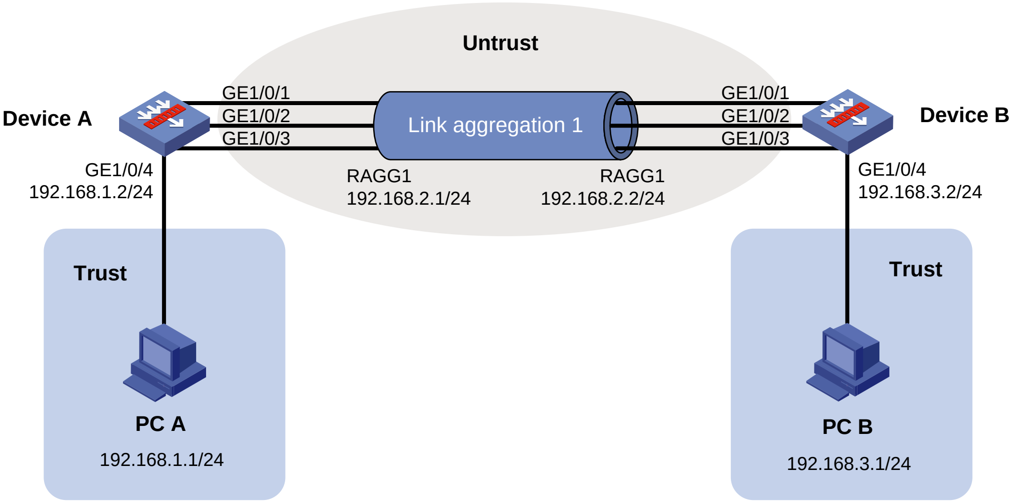
The output shows that link aggregation group 1 is a Layer 3 static aggregation group that contains three Selected ports.

Example: Configuring a Layer 3 dynamic aggregation group

Network configuration

As shown in Figure 9, configure a Layer 3 dynamic aggregation group on both Device A and Device B to aggregate the links between them.

Figure 9 Network diagram



Procedure

1. Configure a dynamic aggregation group:

   # Create Layer 3 aggregate interface Route-Aggregation 1.

   ```
   <DeviceA> system-view
   [DeviceA] interface route-aggregation 1
   ```

   # Assign an IP address to Route-Aggregation 1.

   ```
   [DeviceA-Route-Aggregation1] ip address 192.168.2.1 24
   ```

   # Set the link aggregation mode to dynamic.

   ```
   [DeviceA-Route-Aggregation1] link-aggregation mode dynamic
   [DeviceA-Route-Aggregation1] quit
   ```

   # Assign Layer 3 Ethernet interfaces GigabitEthernet 1/0/1 through GigabitEthernet 1/0/3 to aggregation group 1.

   ```
   [DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/3
   [DeviceA-if-range] port link-aggregation group 1
   [DeviceA-if-range] quit
   ```

2. Assign an IP address to interface GigabitEthernet 1/0/4.

   ```
   [DeviceA] interface GigabitEthernet 1/0/4
   [DeviceA-GigabitEthernet1/0/4] ip address 192.168.1.2 24
   [DeviceA-GigabitEthernet1/0/4] quit
   ```

3. Configure settings for routing.

   This example configure a static route for PC B, and the next hop in the routes is 192.168.2.2.

   ```
   [DeviceA] ip route-static 192.168.3.1 24 192.168.2.2
   ```

4. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/4
   [Device-security-zone-Trust] quit
   [DeviceA] security-zone name untrust
   [Device-security-zone-Untrust] import interface route-aggregation 1
   [Device-security-zone-Untrust] quit
   ```

5. Configure a security policy:

   Configure rules to permit traffic between PC A and PC B:

# Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] source-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-0-trust-untrust] destination-ip-subnet 192.168.3.0 24
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] source-ip-subnet 192.168.3.0 24
[DeviceA-security-policy-ip-1-untrust-trust] destination-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

6. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Creation Mode: Manual
Aggregation Mode: Dynamic
Loadsharing Type: Shar
System ID: 0x8000, 000f-e267-6c6a
Local:
  Port             Status  Priority Oper-Key  Flag
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1         {ACDEF}
  GE1/0/2          S       32768    1         {ACDEF}
  GE1/0/3          S       32768    1         {ACDEF}
Remote:
  Actor            Partner Priority Oper-Key  SystemID               Flag
--------------------------------------------------------------------------------
  GE1/0/1          1       32768    1         0x8000, 000f-e267-57ad {ACDEF}
```

```
GE1/0/2          2        32768     1          0x8000, 000f-e267-57ad {ACDEF}
GE1/0/3          3        32768     1          0x8000, 000f-e267-57ad {ACDEF}
```

The output shows that link aggregation group 1 is a Layer 3 dynamic aggregation group that contains three Selected ports.

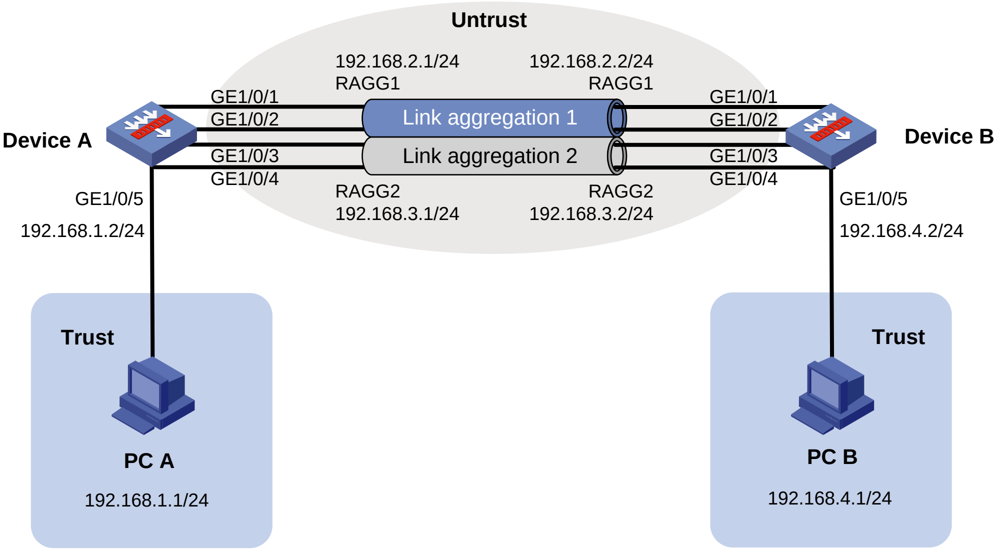
Example: Configuring Layer 3 aggregation load sharing

Network configuration

As shown in Figure 10, perform the following tasks:

- Configure Layer 3 static aggregation groups 1 and 2 on Device A and Device B, respectively.
- Configure link aggregation groups 1 and 2 to load share traffic across aggregation group member ports.
    - Configure link aggregation group 1 to load share packets based on source IP addresses.
    - Configure link aggregation group 2 to load share packets based on destination IP addresses.

Figure 10 Network diagram



Procedure

1. Configure Layer 3 aggregation groups:

# Create Layer 3 aggregate interface Route-Aggregation 1.

```
<DeviceA> system-view
[DeviceA] interface route-aggregation 1
```

# Configure Layer 3 aggregation group 1 to load share packets based on source IP addresses.

```
[DeviceA-Route-Aggregation1] link-aggregation load-sharing mode source-ip
```

# Assign an IP address to Layer 3 aggregate interface Route-Aggregation 1.

```
[DeviceA-Route-Aggregation1] ip address 192.168.2.1 24
[DeviceA-Route-Aggregation1] quit
```

# Assign Layer 3 Ethernet interfaces GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 to aggregation group 1.

```
[DeviceA] interface range gigabitethernet 1/0/1 to gigabitethernet 1/0/2
[DeviceA-if-range] port link-aggregation group 1
[DeviceA-if-range] quit
```

# Create Layer 3 aggregate interface Route-Aggregation 2.

```
[DeviceA] interface route-aggregation 2
```

# Configure Layer 3 aggregation group 2 to load share packets based on destination IP addresses.

```
[DeviceA-Route-Aggregation2] link-aggregation load-sharing mode destination-ip
```

# Assign an IP address to Layer 3 aggregate interface Route-Aggregation 2.

```
[DeviceA-Route-Aggregation2] ip address 192.168.3.1 24
[DeviceA-Route-Aggregation2] quit
```

# Assign Layer 3 Ethernet interfaces GigabitEthernet 1/0/3 and GigabitEthernet 1/0/4 to aggregation group 2.

```
[DeviceA] interface range gigabitethernet 1/0/3 to gigabitethernet 1/0/4
[DeviceA-if-range] port link-aggregation group 2
[DeviceA-if-range] quit
```

2. Assign an IP address to interface GigabitEthernet 1/0/5.

```
[DeviceA] interface gigabitethernet 1/0/5
[DeviceA-GigabitEthernet1/0/5] ip address 192.168.1.2 24
[DeviceA-GigabitEthernet1/0/5] quit
```

3. Configure settings for routing.

This example configure static routes for PC B, and the next hops in the routes are 192.168.2.2 and 192.168.3.2.

```
[DeviceA] ip route-static 192.168.4.1 24 192.168.2.2
[DeviceA] ip route-static 192.168.4.1 24 192.168.3.2
```

4. Add interfaces to security zones.

```
[DeviceA] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/5
[Device-security-zone-Trust] quit
[DeviceA] security-zone name untrust
[Device-security-zone-Untrust] import interface route-aggregation 1
[Device-security-zone-Untrust] import interface route-aggregation 2
[Device-security-zone-Untrust] quit
```

5. Configure a security policy:

Configure rules to permit traffic between PC A and PC B:

# Configure a rule named **trust-untrust** to permit the packets sent from the **Trust** security zone to the **Untrust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-0-trust-untrust] action pass
[DeviceA-security-policy-ip-0-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-0-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-0-trust-untrust] source-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-0-trust-untrust] destination-ip-subnet 192.168.4.0 24
[DeviceA-security-policy-ip-0-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

# Configure a rule named **untrust-trust** to permit the packets sent from the **Untrust** security zone to the **Trust** security zone.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-1-untrust-trust] action pass
```

```
[DeviceA-security-policy-ip-1-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-1-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-1-untrust-trust] source-ip-subnet 192.168.4.0 24
[DeviceA-security-policy-ip-1-untrust-trust] destination-ip-subnet 192.168.1.0 24
[DeviceA-security-policy-ip-1-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

6. Configure Device B in the same way Device A is configured. (Details not shown.)

Verifying the configuration

# Display detailed information about all aggregation groups on Device A.

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Port: A -- Auto port
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1

Aggregate Interface: Route-Aggregation2
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/3          S       32768    2
  GE1/0/4          S       32768    2
```

The output shows that:

- Link aggregation groups 1 and 2 are both load-shared Layer 3 static aggregation groups.
- Each aggregation group contains two Selected ports.

# Display all the group-specific load sharing modes on Device A.

```
[DeviceA] display link-aggregation load-sharing mode interface
Route-Aggregation1 Load-Sharing Mode:
source-ip address

Route-Aggregation2 Load-Sharing Mode:
destination-ip address
```

The output shows that:

- Link aggregation group 1 distributes packets based on source IP addresses.
- Link aggregation group 2 distributes packets based on destination IP addresses.

Contents

Configuring VLANs

About VLANs

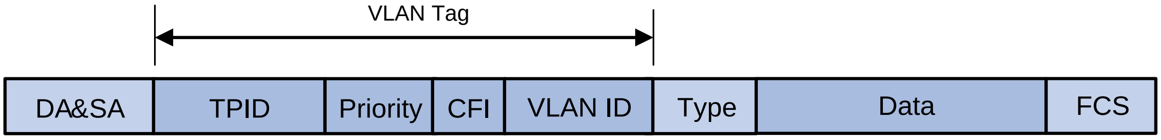
The Virtual Local Area Network (VLAN) technology divides a physical LAN into multiple logical LANs. It has the following benefits:

- **Security**—Hosts in the same VLAN can communicate with one another at Layer 2, but they are isolated from hosts in other VLANs at Layer 2.
- **Broadcast traffic isolation**—Each VLAN is a broadcast domain that limits the transmission of broadcast packets.
- **Flexibility**—A VLAN can be logically divided on a workgroup basis. Hosts in the same workgroup can be assigned to the same VLAN, regardless of their physical locations.

VLAN frame encapsulation

To identify Ethernet frames from different VLANs, IEEE 802.1Q inserts a four-byte VLAN tag between the destination and source MAC address (DA&SA) field and the Type field.

Figure 1 VLAN tag placement and format



A VLAN tag includes the following fields:

- **TPID**—16-bit tag protocol identifier that indicates whether a frame is VLAN-tagged. By default, the hexadecimal TPID value 8100 identifies a VLAN-tagged frame. A device vendor can set the TPID to a different value. For compatibility with a neighbor device, set the TPID value on the device to be the same as the neighbor device. For more information about setting the TPID value, see "Configuring VLAN termination.".
- **Priority**—3-bit long, identifies the 802.1p priority of the frame. For more information, see *ACL and QoS Configuration Guide*.
- **CFI**—1-bit long canonical format indicator that indicates whether the MAC addresses are encapsulated in the standard format when packets are transmitted across different media. Available values include:
    - **0 (default)**—The MAC addresses are encapsulated in the standard format.
    - **1**—The MAC addresses are encapsulated in a non-standard format.

  This field is always set to 0 for Ethernet.
- **VLAN ID**—12-bit long, identifies the VLAN to which the frame belongs. The VLAN ID range is 0 to 4095. VLAN IDs 0 and 4095 are reserved, and VLAN IDs 1 to 4094 are user configurable.

The way a network device handles an incoming frame depends on whether the frame has a VLAN tag and the value of the VLAN tag (if any).

Ethernet supports encapsulation formats Ethernet II, 802.3/802.2 LLC, 802.3/802.2 SNAP, and 802.3 raw. The Ethernet II encapsulation format is used here. For information about the VLAN tag fields in other frame encapsulation formats, see related protocols and standards.

For a frame that has multiple VLAN tags, the device handles it according to its outermost VLAN tag and transmits its inner VLAN tags as the payload.

Port-based VLANs

Port-based VLANs group VLAN members by port. A port forwards packets from a VLAN only after it is assigned to the VLAN.

Port link type

You can set the link type of a port to access, trunk, or hybrid. The port link type determines whether the port can be assigned to multiple VLANs. The link types use the following VLAN tag handling methods:

- **Access**—An access port can forward packets only from one VLAN and send these packets untagged. An access port is typically used in the following conditions:
  - Connecting to a terminal device that does not support VLAN packets.
  - In scenarios that do not distinguish VLANs.
- **Trunk**—A trunk port can forward packets from multiple VLANs. Except packets from the port VLAN ID (PVID), packets sent out of a trunk port are VLAN-tagged. Ports connecting network devices are typically configured as trunk ports.
- **Hybrid**—A hybrid port can forward packets from multiple VLANs. The tagging status of the packets forwarded by a hybrid port depends on the port configuration.

PVID

The PVID identifies the default VLAN of a port. Untagged packets received on a port are considered as the packets from the port PVID.

An access port can join only one VLAN. The VLAN to which the access port belongs is the PVID of the port. A trunk or hybrid port supports multiple VLANs and the PVID configuration.

How ports of different link types handle frames

<table>
<tr><th>Actions</th><th>Access</th><th>Trunk</th><th>Hybrid</th></tr>
<tr><td>In the inbound direction for an untagged frame</td><td>Tags the frame with the PVID tag.</td><td colspan="2">• If the PVID is permitted on the port, tags the frame with the PVID tag.
• If not, drops the frame.</td></tr>
<tr><td>In the inbound direction for a tagged frame</td><td>• Receives the frame if its VLAN ID is the same as the PVID.
• Drops the frame if its VLAN ID is different from the PVID.</td><td colspan="2">• Receives the frame if its VLAN is permitted on the port.
• Drops the frame if its VLAN is not permitted on the port.</td></tr>
<tr><td>In the outbound direction</td><td>Removes the VLAN tag and sends the frame.</td><td>• Removes the tag and sends the frame if the frame carries the PVID tag and the port belongs to the PVID.
• Sends the frame without removing the tag if its VLAN is carried on the port but is different from the PVID.</td><td>Sends the frame if its VLAN is permitted on the port. The tagging status of the frame depends on the <b>port hybrid vlan</b> command configuration.</td></tr>
</table>

Layer 3 communication between VLANs

Hosts of different VLANs use VLAN interfaces to communicate at Layer 3. VLAN interfaces are virtual interfaces that do not exist as physical entities on devices. For each VLAN, you can create one VLAN interface and assign an IP address to it. The VLAN interface acts as the gateway of the VLAN to forward packets destined for another IP subnet at Layer 3.

Protocols and standards

IEEE 802.1Q, *IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks*

Configuring a VLAN

Restrictions and guidelines

- As the system default VLAN, VLAN 1 cannot be created or deleted.
- Before you delete a dynamic VLAN or a VLAN locked by an application, you must first remove the configuration from the VLAN.

Creating VLANs

1. Enter system view.

   `system-view`

2. Create one or multiple VLANs.
   - Create a VLAN and enter its view.

     `vlan` *`vlan-id`*
   - Create multiple VLANs and enter VLAN view.

     Create VLANs.

     `vlan` { *`vlan-id1`* **`to`** *`vlan-id2`* | **`all`** }

     Enter VLAN view.

     `vlan` *`vlan-id`*

   By default, only the system default VLAN (VLAN 1) exists.

3. (Optional.) Set a name for the VLAN.

   `name` *`text`*

   By default, the name of a VLAN is **VLAN** *vlan-id.* The *vlan-id* argument specifies the VLAN ID in a four-digit format. If the VLAN ID has fewer than four digits, leading zeros are added. For example, the name of VLAN 100 is **VLAN 0100**.

4. (Optional.) Configure the description for the VLAN.

   `description` *`text`*

   By default, the description of a VLAN is **VLAN** *vlan-id.* The *vlan-id* argument specifies the VLAN ID in a four-digit format. If the VLAN ID has fewer than four digits, leading zeros are added. For example, the default description of VLAN 100 is **VLAN 0100**.

Configuring port-based VLANs

Restrictions and guidelines for port-based VLANs

- When you use the **undo vlan** command to delete the PVID of a port, either of the following events occurs depending on the port link type:
  - For an access port, the PVID of the port changes to VLAN 1.
  - For a hybrid or trunk port, the PVID setting of the port does not change.

  You can use a nonexistent VLAN as the PVID for a hybrid or trunk port, but not for an access port.
- As a best practice, set the same PVID for a local port and its peer.
- To prevent a port from dropping untagged packets or PVID-tagged packets, assign the port to its PVID.

Assigning an access port to a VLAN

About this task

You can assign an access port to a VLAN in VLAN view or interface view.

Assigning one or multiple access ports to a VLAN in VLAN view

1. Enter system view.

   system-view
2. Enter VLAN view.

   vlan *vlan-id*
3. Assign one or multiple access ports to the VLAN.

   port *interface-list*

   By default, all ports belong to VLAN 1.

Assigning an access port to a VLAN in interface view

1. Enter system view.

   system-view
2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     interface *interface-type interface-number*
   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*
3. Set the port link type to access.

   port link-type access

   By default, all ports are access ports.
4. Assign the access port to a VLAN.

   port access vlan *vlan-id*

   By default, all access ports belong to VLAN 1.

Assigning a trunk port to a VLAN

About this task

A trunk port supports multiple VLANs. You can assign it to a VLAN in interface view.

Restrictions and guidelines

To change the link type of a port from trunk to hybrid, set the link type to access first.

To enable a trunk port to transmit packets from its PVID, you must assign the trunk port to the PVID by using the **`port trunk permit vlan`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     `interface` *`interface-type interface-number`*
   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*

3. Set the port link type to trunk.

   `port link-type trunk`

   By default, all ports are access ports.

4. Assign the trunk port to the specified VLANs.

   `port trunk permit vlan` { *`vlan-id-list`* | **`all`** }

   By default, a trunk port permits only VLAN 1.

5. (Optional.) Set the PVID for the trunk port.

   `port trunk pvid vlan` *`vlan-id`*

   The default setting is VLAN 1.

Assigning a hybrid port to a VLAN

About this task

A hybrid port supports multiple VLANs. You can assign it to the specified VLANs in interface view. Make sure the VLANs have been created.

Restrictions and guidelines

To change the link type of a port from trunk to hybrid, set the link type to access first.

To enable a hybrid port to transmit packets from its PVID, you must assign the hybrid port to the PVID by using the **`port hybrid vlan`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     `interface` *`interface-type interface-number`*
   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*

3. Set the port link type to hybrid.

   port link-type hybrid

   By default, all ports are access ports.

4. Assign the hybrid port to the specified VLANs.

   port hybrid vlan *vlan-id-list* { **tagged** | **untagged** }

   By default, the hybrid port is an untagged member of the VLAN to which the port belongs when its link type is **access**.

5. (Optional.) Set the PVID for the hybrid port.

   port hybrid pvid vlan *vlan-id*

   By default, the PVID of a hybrid port is the ID of the VLAN to which the port belongs when its link type is **access**.

Configuring a VLAN group

About this task

A VLAN group includes a set of VLANs.

Procedure

1. Enter system view.

   system-view

2. Create a VLAN group and enter its view.

   vlan-group *group-name*

3. Add VLANs to the VLAN group.

   vlan-list *vlan-id-list*

   By default, no VLANs exist in a VLAN group.

   You can add multiple VLAN lists to a VLAN group.

Configuring VLAN interfaces

VLAN interfaces configuration tasks at a glance

To configure VLAN interfaces, perform the following tasks:

1. Creating a VLAN interface
2. (Optional.) Restoring the default settings for the VLAN interface

Prerequisites

Before you create a VLAN interface for a VLAN, create the VLAN first.

Creating a VLAN interface

1. Enter system view.

   system-view

2. Create a VLAN interface and enter its view.

   interface vlan-interface *interface-number*

3. Assign an IP address to the VLAN interface.

   ip address *ip-address* { *mask* | *mask-length* } [**sub**]

   By default, no IP address is assigned to a VLAN interface.

4. (Optional.) Configure the description for the VLAN interface.

   description *text*

   The default setting is the VLAN interface name. For example, **Vlan-interface1 Interface**.

5. (Optional.) Set the MTU for the VLAN interface.

   mtu *size*

   By default, the MTU is 1500 for a VLAN interface.

6. (Optional.) Set the expected bandwidth for the interface.

   bandwidth *bandwidth-value*

   By default, the expected bandwidth (in kbps) is the interface baud rate divided by 1000.

7. Bring up the VLAN interface.

   undo shutdown

   By default, a VLAN interface is not manually shut down.

Restoring the default settings for the VLAN interface

Restrictions and guidelines

This feature might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **display this** command in interface view to identify these commands, and then use their **undo** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.

   system-view

2. Enter a VLAN interface view.

   interface vlan-interface *interface-number*

3. Restore the default settings for the VLAN interface.

   default

△ **CAUTION:**

This feature might interrupt ongoing network services. Make sure you are fully aware of the impact of this feature when you use it on a live network.

Display and maintenance commands for VLANs

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display VLAN interface information.	**display interface** [**vlan-interface** [*interface-number*]] [**brief**]
Display hybrid ports or trunk ports on the	**display port** { **hybrid** | **trunk** }

Task	Command
device.	
Display VLAN information.	**display vlan** [*vlan-id1* [**to** *vlan-id2*] | **all** | **dynamic** | **static**]
Display brief VLAN information.	**display vlan brief**
Display VLAN group information.	**display vlan-group** [*group-name*]
Clear statistics on a port.	**reset counters interface** [**vlan-interface** [*interface-number*]]

Contents

Configuring VLAN termination

About VLAN termination

VLAN termination typically processes packets that include VLAN tags. A VLAN termination-enabled interface performs the following tasks when receiving a VLAN-tagged packet:

1. Assigns the packet to an interface according to its VLAN tags.
2. Removes the VLAN tags of the packet.
3. Delivers the packet to Layer 3 forwarding or other processing pipelines.

Before sending the packet, the VLAN termination-enabled interface determines whether to add new VLAN tags to the packet, based on the VLAN termination type.

VLAN termination can also process packets that do not include any VLAN tags.

This document uses the following VLAN tag concepts for a packet that has two or more layers of VLAN tags:

- **Layer 1 VLAN tag**—Specifies the outermost layer of VLAN tags.
- **Layer 2 VLAN tag**—Specifies the second outermost layer of VLAN tags.

The VLAN IDs of the packets are numbered in the same manner as the VLAN tags.

VLAN termination types

VLAN termination types	Types of packets to be terminated on the interface	Tags of outgoing packets on the interface
Dot1q termination	The packets must meet both of the following requirements: • The packets include one or more layers of VLAN tags. • The outermost VLAN ID matches the configured value.	Single-tagged
Untagged termination	Untagged packets.	Untagged
Default termination	Packets that cannot be processed on any other subinterfaces of the same main interface.	Untagged

VLAN termination mechanism

VLAN interfaces and subinterfaces, such as Layer 3 Ethernet subinterfaces and Layer 3 aggregate subinterfaces, can terminate the following packets:

- Packets whose outermost VLAN IDs match the configured values.
- Packets whose outermost two layers of VLAN IDs match the configured values.

A VLAN interface terminates only the packets whose outermost VLAN ID is the same as the VLAN interface number. For example, VLAN-interface 10 terminates only the packets with the outermost VLAN tag 10.

A main interface does not terminate VLAN-tagged packets (for example, Layer 3 Ethernet interface or Layer 3 aggregate interface). To terminate VLAN-tagged packets, create subinterfaces for the main interface.

Subinterfaces of the same main interface can use different types of VLAN termination. To process received packets, the system selects a subinterface based on the following VLAN termination types in descending order of priority:

- Dot1q termination or support for Dot1q termination by default.
- Untagged termination.
- Default termination.

If none of these VLAN termination types applies, the main interface processes the packets.

If untagged termination is enabled on a subinterface of an interface, untagged packets are processed by the subinterface instead of the main interface.

If default termination is enabled on a subinterface of an interface, packets are processed by the subinterface instead of the main interface.

When a main interface is bound to a VLAN interface, the main interface processes VLAN-tagged packets according to the VLAN termination configuration of the VLAN interface.

VLAN termination application scenarios

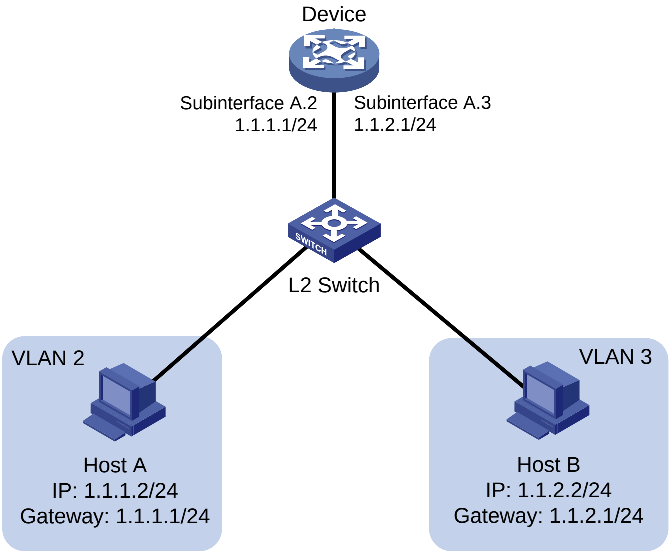
Inter-VLAN communication

Hosts in different VLANs cannot directly communicate with each other. You can use Layer 3 routing to allow all VLANs to communicate. To restrict communication to the specified VLANs, configure VLAN termination on subinterfaces or VLAN interfaces.

As shown in Figure 1, Host A and Host B are in different VLANs. For the two hosts to communicate with each other, perform the following tasks:

1. Specify 1.1.1.1/24 and 1.1.2.1/24 as the gateway IP address for Host A and Host B, respectively.
2. On the device, configure Dot1q termination on Layer 3 Ethernet subinterfaces Subinterface A.2 and Subinterface A.3.

Figure 1 Inter-VLAN communication through Layer 3 subinterfaces



As shown in Figure 2, Host A is in VLAN 2, Host B is in VLAN 3, and Host C is in VLAN 4. For Host A and Host B to communicate with each other, perform the following tasks:

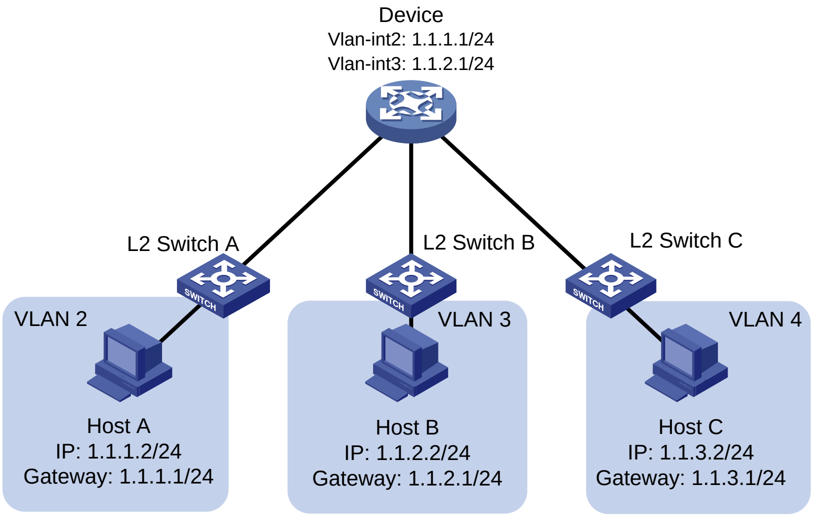
1. Specify 1.1.1.1/24 and 1.1.2.1/24 as the gateway IP address for Host A and Host B, respectively.

2. On the device, create VLAN-interface 2 and configure the IP address as 1.1.1.1/24, which is the same as the gateway address of Host A. Create VLAN-interface 3 and configure the IP address as 1.1.2.1/24, which is the same as the gateway address of Host B.

VLAN termination by the outermost VLAN ID of packets is automatically performed on VLAN interfaces. For example, when Host A sends a packet to Host B, VLAN-interface 2 removes the VLAN tag from the packet and forwards it to VLAN-interface 3. Then, VLAN-interface 3 tags the packet with VLAN 3 and Host B can receive the packet.

Because the device does not have a VLAN interface to terminate packets from VLAN 4, Host C cannot communicate with Host A or Host B.

Figure 2 Inter-VLAN communication through VLAN interfaces



Restrictions and guidelines: VLAN termination configuration

When you configure VLAN termination, follow these restrictions and guidelines:

- On a portal-enabled interface, log off all portal users before you change the VLAN termination type. Any portal users who remain online after the change cannot be logged off or reauthenticated. For more information about portal authentication, see *Security Configuration Guide*.
- After you modify the VLAN termination configuration for a subinterface, the subinterface automatically restarts. All dynamic ARP table entries for the subinterface are deleted.

VLAN termination tasks at a glance

To configure VLAN termination, perform the following tasks:

1. (Required.) Configuring VLAN termination

   Choose one of the following tasks:

   - Configuring ambiguous Dot1q termination
   - Configuring unambiguous Dot1q termination
   - Configuring untagged termination
   - Configuring default termination

2. (Optional.) Enabling a VLAN termination-enabled interface to transmit broadcasts and multicasts

   Perform this task to enable VLAN termination-enabled interfaces to transmit broadcasts and multicasts.
3. (Optional.) Configuring the TPID for VLAN-tagged packets

Configuring ambiguous Dot1q termination

About this task

Use this feature to terminate VLAN-tagged packets whose outermost VLAN IDs are in the specified range. Other VLAN-tagged packets are not allowed to pass.

When an interface receives a packet, it removes the outermost VLAN ID from the packet. When the interface sends a packet, it tags the packet with a VLAN ID as follows:

- For a PPPoE packet, the VLAN ID is from the matching PPPoE session entry.
- For a DHCP relay packet, the VLAN ID is from the matching DHCP session entry.

Procedure

1. Enter system view.

   system-view
2. Enter interface view.
   - Enter Layer 3 Ethernet subinterface view.

     interface *interface-type interface-number.subnumber*
   - Enter Layer 3 aggregate subinterface view.

     interface route-aggregation *interface-number.subnumber*
   - Enter Reth subinterface view.

     interface reth *interface-number.subnumber*
3. Configure ambiguous Dot1q termination.

   vlan-type dot1q vid *vlan-id-list*

   By default, Dot1q termination is disabled on a subinterface.

Configuring unambiguous Dot1q termination

About this task

Use this feature to terminate only VLAN-tagged packets whose outermost VLAN ID matches the specified VLAN ID. Other VLAN-tagged packets are not allowed to pass.

When an interface receives a packet, it removes the outermost VLAN ID from the packet. When the interface sends a packet, it tags the packet with the specified VLAN ID.

Procedure

1. Enter system view.

   system-view
2. Enter interface view.
   - Enter Layer 3 Ethernet subinterface view.

     interface *interface-type interface-number.subnumber*
   - Enter Layer 3 aggregate subinterface view.

     interface route-aggregation *interface-number.subnumber*

    - Enter Reth subinterface view.

      interface reth *interface-number.subnumber*

3. Configure unambiguous Dot1q termination.

   vlan-type dot1q vid *vlan-id*

   By default, Dot1q termination is disabled on a subinterface.

Configuring untagged termination

1. Enter system view.

   system-view

2. Enter interface view.
    - Enter Layer 3 Ethernet subinterface view.

      interface *interface-type interface-number.subnumber*
    - Enter Layer 3 aggregate subinterface view.

      interface route-aggregation *interface-number.subnumber*
    - Enter Reth subinterface view.

      interface reth *interface-number.subnumber*
3. Configure untagged termination.

   vlan-type dot1q untagged

   By default, untagged termination is disabled on a subinterface.

Configuring default termination

1. Enter system view.

   system-view

2. Enter interface view.
    - Enter Layer 3 Ethernet subinterface view.

      interface *interface-type interface-number.subnumber*
    - Enter Layer 3 aggregate subinterface view.

      interface route-aggregation *interface-number.subnumber*
    - Enter Reth subinterface view.

      interface reth *interface-number.subnumber*
3. Configure default termination.

   vlan-type dot1q default

   By default, default termination is disabled on a subinterface.

Enabling a VLAN termination-enabled interface to transmit broadcasts and multicasts

About this task

This function enables ambiguous Dot1q termination-enabled interfaces to transmit broadcasts and multicasts.

To transmit a broadcast or multicast packet, the interface starts a traversal over the VLAN IDs specified for ambiguous termination. It copies the packet and tags each copy with a VLAN ID, until all VLAN IDs in the specified range are traversed.

Restrictions and guidelines

As a best practice, use the **vlan-termination broadcast ra** command to enable an ambiguous Dot1q termination-enabled interface to transmit RA multicast packets on an IPv6 network. This command prohibits transmission of broadcast packets and other types of multicast packets, and consumes less CPU resources than the **vlan-termination broadcast enable** command.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   - Enter Layer 3 Ethernet subinterface view.
     interface *interface-type interface-number.subnumber*
   - Enter Layer 3 aggregate subinterface view.
     interface route-aggregation *interface-number.subnumber*
   - Enter Reth subinterface view.
     interface reth *interface-number.subnumber*
3. Enable the interface to transmit broadcasts and multicasts.
   - Enable the interface to transmit broadcasts and multicasts.
     vlan-termination broadcast enable
   - Enable the interface to transmit only RA multicasts on an IPv6 network.
     vlan-termination broadcast ra

   By default, an ambiguous Dot1q termination-enabled interface does not transmit broadcasts and multicasts.

Configuring the TPID for VLAN-tagged packets

About this task

TPID identifies whether or not a frame contains VLAN tags. By default, the value of 0x8100 identifies an IEEE 802.1Q-tagged frame. You can set another TPID value to identify VLAN-tagged packets.

To work with VLAN termination on a subinterface, set the TPID value in the outermost VLAN tag of packets on the main interface of the subinterface. If VLAN termination is enabled on a VLAN interface, set the TPID value in the outermost VLAN tag of packets on the same VLAN interface.

The interface processes packets as untagged packets if their outermost VLAN tag is not 0x8100 or the configured value.

When sending a packet, the interface sets the TPID value in the outermost VLAN tag to the configured value. If the packet includes two or more layers of VLAN tags, the interface sets the TPID values to 0x8100 in all VLAN tags except the outermost VLAN tag.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   - Enter Layer 3 Ethernet interface view.

    interface interface-*type* *interface-number*

    - Enter Layer 3 aggregate interface view.

      interface route-aggregation *interface-number*

    - Enter Reth subinterface view.

      interface reth *interface-number.subnumber*

3. Set the TPID value in the outermost VLAN tag of packets received and sent by the interface.

   dot1q ethernet-type *hex-value*

   The default setting is 0x8100.

Contents

Spanning tree protocol overview

Spanning tree protocols eliminate loops in a physical link-redundant network by selectively blocking redundant links and putting them in a standby state.

The recent versions of STP include the Rapid Spanning Tree Protocol (RSTP), the Per-VLAN Spanning Tree (PVST), and the Multiple Spanning Tree Protocol (MSTP).

About STP

STP was developed based on the 802.1d standard of IEEE to eliminate loops at the data link layer in a LAN. Networks often have redundant links as backups in case of failures, but loops are a very serious problem. Devices running STP detect loops in the network by exchanging information with one another. They eliminate loops by selectively blocking certain ports to prune the loop structure into a loop-free tree structure. This avoids proliferation and infinite cycling of packets that would occur in a loop network.

In a narrow sense, STP refers to IEEE 802.1d STP. In a broad sense, STP refers to the IEEE 802.1d STP and various enhanced spanning tree protocols derived from that protocol.

STP protocol frames

STP uses bridge protocol data units (BPDUs), also known as configuration messages, as its protocol frames. This chapter uses BPDUs to represent all types of spanning tree protocol frames.

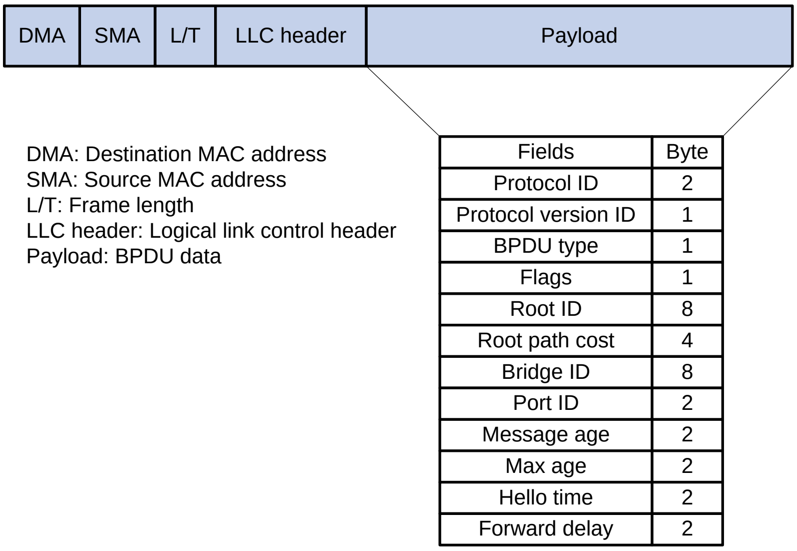
STP-enabled devices exchange BPDUs to establish a spanning tree. BPDUs contain sufficient information for the devices to complete spanning tree calculation.

STP uses two types of BPDUs, configuration BPDUs and topology change notification (TCN) BPDUs.

Configuration BPDUs

Devices exchange configuration BPDUs to elect the root bridge and determine port roles. Figure 1 shows the configuration BPDU format.

Figure 1 Configuration BPDU format



DMA: Destination MAC address
SMA: Source MAC address
L/T: Frame length
LLC header: Logical link control header
Payload: BPDU data

Fields	Byte
Protocol ID	2
Protocol version ID	1
BPDU type	1
Flags	1
Root ID	8
Root path cost	4
Bridge ID	8
Port ID	2
Message age	2
Max age	2
Hello time	2
Forward delay	2

The payload of a configuration BPDU includes the following fields:

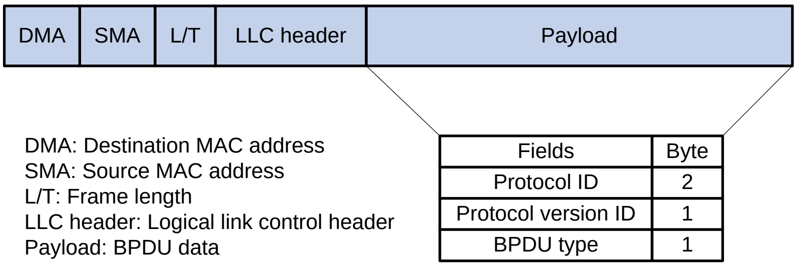
- **Protocol ID**—Fixed at 0x0000, which represents IEEE 802.1d.
- **Protocol version ID**—Spanning tree protocol version ID. The protocol version ID for STP is 0x00.
- **BPDU type**—Type of the BPDU. The value is 0x00 for a configuration BPDU.
- **Flags**—An 8-bit field indicates the purpose of the BPDU. The lowest bit is the Topology Change (TC) flag. The highest bit is the Topology Change Acknowledge (TCA) flag. All other bits are reserved.
- **Root ID**—Root bridge ID formed by the priority and MAC address of the root bridge.
- **Root path cost**—Cost of the path to the root bridge.
- **Bridge ID**—Designated bridge ID formed by the priority and MAC address of the designated bridge.
- **Port ID**—Designated port ID formed by the priority and global port number of the designated port.
- **Message age**—Age of the configuration BPDU while it propagates in the network.
- **Max age**—Maximum age of the configuration BPDU stored on the switch.
- **Hello time**—Configuration BPDU transmission interval.
- **Forward delay**—Delay for STP bridges to transit port state.

Devices use the root bridge ID, root path cost, designated bridge ID, designated port ID, message age, max age, hello time, and forward delay for spanning tree calculation.

TCN BPDUs

Devices use TCN BPDUs to announce changes in the network topology. Figure 2 shows the TCN BPDU format.

Figure 2 TCN BPDU format



DMA: Destination MAC address
SMA: Source MAC address
L/T: Frame length
LLC header: Logical link control header
Payload: BPDU data

Fields	Byte
Protocol ID	2
Protocol version ID	1
BPDU type	1

The payload of a TCN BPDU includes the following fields:

- **Protocol ID**—Fixed at 0x0000, which represents IEEE 802.1d.
- **Protocol version ID**—Spanning tree protocol version ID. The protocol version ID for STP is 0x00.
- **BPDU type**—Type of the BPDU. The value is 0x80 for a TCN BPDU.

A non-root bridge sends TCN BPDUs when one of the following events occurs on the bridge:

- A port transits to the forwarding state, and the bridge has a minimum of one designated port.
- A port transits from the forwarding or learning state to the blocking state.

The non-root bridge uses TCN BPDUs to notify the root bridge once the network topology changes. The root bridge then sets the TC flag in its configuration BPDU and propagates it to other bridges.

Basic concepts in STP

Root bridge

A tree network must have a root bridge. The entire network contains only one root bridge, and all the other bridges in the network are called leaf nodes. The root bridge is not permanent, but can change with changes of the network topology.

Upon initialization of a network, each device generates and periodically sends configuration BPDUs, with itself as the root bridge. After network convergence, only the root bridge generates and periodically sends configuration BPDUs. The other devices only forward the BPDUs.

Root port

On a non-root bridge, the port nearest to the root bridge is the root port. The root port communicates with the root bridge. Each non-root bridge has only one root port. The root bridge has no root port.

Designated bridge and designated port

Classification	Designated bridge	Designated port
For a device	Device directly connected to the local device and responsible for forwarding BPDUs to the local device.	Port through which the designated bridge forwards BPDUs to this device.
For a LAN	Device responsible for forwarding BPDUs to this LAN segment.	Port through which the designated bridge forwards BPDUs to this LAN segment.

As shown in Figure 3, Device B and Device C are directly connected to a LAN.

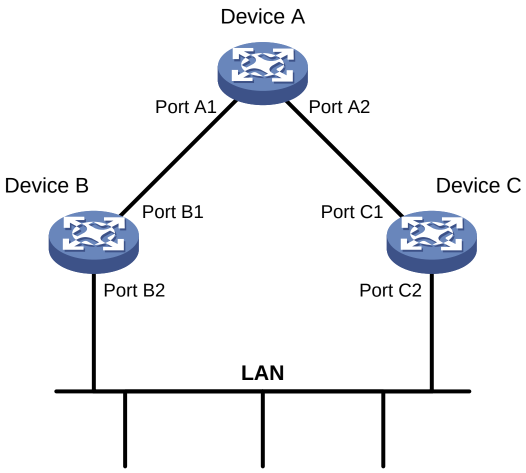
If Device A forwards BPDUs to Device B through port A1, the designated bridge and designated port are as follows:

- The designated bridge for Device B is Device A.
- The designated port for Device B is port A1 on Device A.

If Device B forwards BPDUs to the LAN, the designated bridge and designated port are as follows:

- The designated bridge for the LAN is Device B.
- The designated port for the LAN is port B2 on Device B.

Figure 3 Designated bridges and designated ports



Port states

Table 1 lists the port states in STP.

Table 1 STP port states

State	Receives/sends BPDUs	Learns MAC addresses	Forwards user data
Disabled	No	No	No
Listening	Yes	No	No
Learning	Yes	Yes	No
Forwarding	Yes	Yes	Yes
Blocking	Receive	No	No

Path cost

Path cost is a reference value used for link selection in STP. To prune the network into a loop-free tree, STP calculates path costs to select the most robust links and block redundant links that are less robust.

Calculation process of the STP algorithm

In STP calculation, a device compares the priorities of the received configuration BPDUs from different ports, and elects the root bridge, root ports and designated ports. When the spanning tree calculation is completed, a tree-shape topology forms.

The spanning tree calculation process described in the following sections is an example of a simplified process.

Network initialization

Upon initialization of a device, each port generates a BPDU with the following contents:

- The port as the designated port.
- The device as the root bridge.
- 0 as the root path cost.
- The device ID as the designated bridge ID.

Root bridge selection

The root bridge can be selected in the following methods:

- **Automatic election**—Initially, each STP-enabled device on the network assumes itself to be the root bridge, with its own device ID as the root bridge ID. By exchanging configuration BPDUs, the devices compare their root bridge IDs to elect the device with the smallest root bridge ID as the root bridge.
- **Manual assignment**—You can configure a device as the root bridge or a secondary root bridge of a spanning tree.
    - A spanning tree can have only one root bridge. If you configure multiple devices as the root bridge for a spanning tree, the device with the lowest MAC address is selected.
    - You can configure one or multiple secondary root bridges for a spanning tree. When the root bridge fails or is shut down, a secondary root bridge can take over. If multiple secondary root bridges are configured, the one with the lowest MAC address is selected. However, if a new root bridge is configured, the secondary root bridge is not selected.

Root port and designated ports selection on the non-root bridges

Step	Description
1	A non-root-bridge device regards the port on which it received the optimum configuration BPDU as the root port. Table 2 describes how the optimum configuration BPDU is selected.

Step	Description
2	Based on the configuration BPDU and the path cost of the root port, the device calculates a designated port configuration BPDU for each of the other ports. • The root bridge ID is replaced with that of the configuration BPDU of the root port. • The root path cost is replaced with that of the configuration BPDU of the root port plus the path cost of the root port. • The designated bridge ID is replaced with the ID of this device. • The designated port ID is replaced with the ID of this port.
3	The device compares the calculated configuration BPDU with the configuration BPDU on the port whose port role will be determined. Then, the device acts depending on the result of the comparison: • If the calculated configuration BPDU is superior, the device performs the following operations: ○ Considers this port as the designated port. ○ Replaces the configuration BPDU on the port with the calculated configuration BPDU. ○ Periodically sends the calculated configuration BPDU. • If the configuration BPDU on the port is superior, the device blocks this port without updating its configuration BPDU. The blocked port can receive BPDUs, but cannot send BPDUs or forward data traffic.

When the network topology is stable, only the root port and designated ports forward user traffic. Other ports are all in the blocking state to receive BPDUs but not to forward BPDUs or user traffic.

Table 2 Selecting the optimum configuration BPDU

Step	Actions
1	Upon receiving a configuration BPDU on a port, the device compares the priority of the received configuration BPDU with that of the configuration BPDU generated by the port. • If the former priority is lower, the device discards the received configuration BPDU and keeps the configuration BPDU the port generated. • If the former priority is higher, the device replaces the content of the configuration BPDU generated by the port with the content of the received configuration BPDU.
2	The device compares the configuration BPDUs of all the ports and chooses the optimum configuration BPDU.

The following are the principles of configuration BPDU comparison:

1. The configuration BPDU with the lowest root bridge ID has the highest priority.
2. If configuration BPDUs have the same root bridge ID, their root path costs are compared. For example, the root path cost in a configuration BPDU plus the path cost of a receiving port is S. The configuration BPDU with the smallest S value has the highest priority.
3. If all configuration BPDUs have the same root bridge ID and S value, the following attributes are compared in sequence:
    a. Designated bridge IDs.
    b. Designated port IDs.
    c. IDs of the receiving ports.

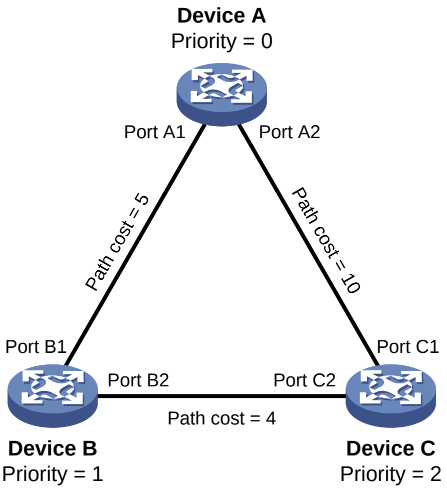
    The configuration BPDU that contains a smaller designated bridge ID, designated port ID, or receiving port ID is selected.

A tree-shape topology forms when the root bridge, root ports, and designated ports are selected.

Example of STP calculation

Figure 4 provides an example showing how the STP algorithm works.

Figure 4 The STP algorithm



As shown in Figure 4, the priority values of Device A, Device B, and Device C are 0, 1, and 2, respectively. The path costs of links among the three devices are 5, 10, and 4.

Device state initialization

In Table 3, each configuration BPDU contains the following fields: root bridge ID, root path cost, designated bridge ID, and designated port ID.

Table 3 Initial state of each device

Device	Port name	Configuration BPDU on the port
Device A	Port A1	{0, 0, 0, Port A1}
	Port A2	{0, 0, 0, Port A2}
Device B	Port B1	{1, 0, 1, Port B1}
	Port B2	{1, 0, 1, Port B2}
Device C	Port C1	{2, 0, 2, Port C1}
	Port C2	{2, 0, 2, Port C2}

Configuration BPDUs comparison on each device

In Table 4, each configuration BPDU contains the following fields: root bridge ID, root path cost, designated bridge ID, and designated port ID.

Table 4 Comparison process and result on each device

Device	Comparison process	Configuration BPDU on ports after comparison
Device A	Port A1 performs the following operations: **1.** Receives the configuration BPDU of Port B1 {1, 0, 1, Port B1}. **2.** Determines that its existing configuration BPDU {0, 0, 0, Port A1} is superior to the received configuration BPDU. **3.** Discards the received one. Port A2 performs the following operations: **1.** Receives the configuration BPDU of Port C1 {2, 0, 2, Port C1}. **2.** Determines that its existing configuration BPDU {0, 0, 0, Port A2} is superior to the received configuration BPDU. **3.** Discards the received one. Device A determines that it is both the root bridge and designated bridge in the configuration BPDUs of all its ports. It considers itself as the root bridge. It does not change the configuration BPDU of any port and starts to periodically send configuration BPDUs.	• Port A1: {0, 0, 0, Port A1} • Port A2: {0, 0, 0, Port A2}
Device B	Port B1 performs the following operations: **4.** Receives the configuration BPDU of Port A1 {0, 0, 0, Port A1}. **5.** Determines that the received configuration BPDU is superior to its existing configuration BPDU {1, 0, 1, Port B1}. **6.** Updates its configuration BPDU. Port B2 performs the following operations: **1.** Receives the configuration BPDU of Port C2 {2, 0, 2, Port C2}. **2.** Determines that its existing configuration BPDU {1, 0, 1, Port B2} is superior to the received configuration BPDU. **3.** Discards the received BPDU.	• Port B1: {0, 0, 0, Port A1} • Port B2: {1, 0, 1, Port B2}
	Device B performs the following operations: **1.** Compares the configuration BPDUs of all its ports. **2.** Decides that the configuration BPDU of Port B1 is the optimum. **3.** Selects Port B1 as the root port with the configuration BPDU unchanged. Based on the configuration BPDU and path cost of the root port, Device B calculates a designated port configuration BPDU for Port B2 {0, 5, 1, Port B2}. Device B compares it with the existing configuration BPDU of Port B2 {1, 0, 1, Port B2}. Device B determines that the calculated one is superior, and determines that Port B2 is the designated port. It replaces the configuration BPDU on Port B2 with the calculated one, and periodically sends the calculated configuration BPDU.	• Root port (Port B1): {0, 0, 0, Port A1} • Designated port (Port B2): {0, 5, 1, Port B2}
Device C	Port C1 performs the following operations: **1.** Receives the configuration BPDU of Port A2 {0, 0, 0, Port A2}. **2.** Determines that the received configuration BPDU is superior to its existing configuration BPDU {2, 0, 2,	• Port C1: {0, 0, 0, Port A2} • Port C2: {1, 0, 1, Port B2}

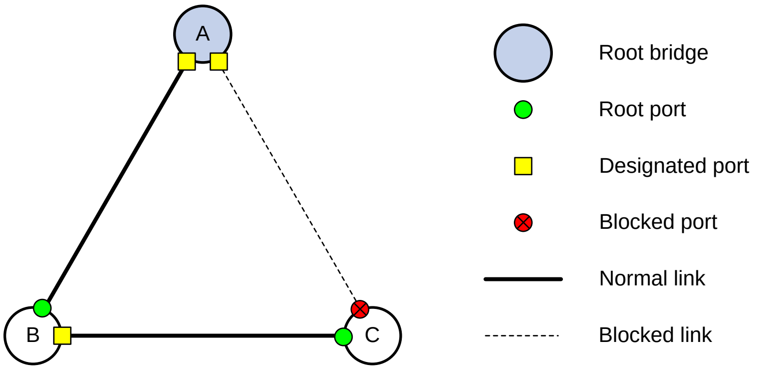
Device	Comparison process	Configuration BPDU on ports after comparison
	Port C1}. **3.** Updates its configuration BPDU. Port C2 performs the following operations: **1.** Receives the original configuration BPDU of Port B2 {1, 0, 1, Port B2}. **2.** Determines that the received configuration BPDU is superior to the existing configuration BPDU {2, 0, 2, Port C2}. **3.** Updates its configuration BPDU.	
	Device C performs the following operations: **1.** Compares the configuration BPDUs of all its ports. **2.** Decides that the configuration BPDU of Port C1 is the optimum. **3.** Selects Port C1 as the root port with the configuration BPDU unchanged. Based on the configuration BPDU and path cost of the root port, Device C calculates the configuration BPDU of Port C2 {0, 10, 2, Port C2}. Device C compares it with the existing configuration BPDU of Port C2 {1, 0, 1, Port B2}. Device C determines that the calculated configuration BPDU is superior to the existing one, selects Port C2 as the designated port, and replaces the configuration BPDU of Port C2 with the calculated one.	• Root port (Port C1): {0, 0, 0, Port A2} • Designated port (Port C2): {0, 10, 2, Port C2}
	Port C2 performs the following operations: **1.** Receives the updated configuration BPDU of Port B2 {0, 5, 1, Port B2}. **2.** Determines that the received configuration BPDU is superior to its existing configuration BPDU {0, 10, 2, Port C2}. **3.** Updates its configuration BPDU. Port C1 performs the following operations: **1.** Receives a periodic configuration BPDU {0, 0, 0, Port A2} from Port A2. **2.** Determines that it is the same as the existing configuration BPDU. **3.** Discards the received BPDU.	• Port C1: {0, 0, 0, Port A2} • Port C2: {0, 5, 1, Port B2}
	Device C determines that the root path cost of Port C1 is larger than that of Port C2. The root path cost of Port C1 is 10, root path cost of the received configuration BPDU (0) plus path cost of Port C1 (10). The root path cost of Port C2 is 9, root path cost of the received configuration BPDU (5) plus path cost of Port C2 (4). Device C determines that the configuration BPDU of Port C2 is the optimum, and selects Port C2 as the root port with the configuration BPDU unchanged. Based on the configuration BPDU and path cost of the root port, Device C performs the following operations: **1.** Calculates a designated port configuration BPDU for Port C1 {0, 9, 2, Port C1}. **2.** Compares it with the existing configuration BPDU of Port C1 {0, 0, 0, Port A2}. **3.** Determines that the existing configuration BPDU is superior to the calculated one and blocks Port C1 with the configuration BPDU unchanged. Port C1 does not forward data until a new event triggers a	• Blocked port (Port C1): {0, 0, 0, Port A2} • Root port (Port C2): {0, 5, 1, Port B2}

Device	Comparison process	Configuration BPDU on ports after comparison
	spanning tree calculation process: for example, the link between Device B and Device C is down.	

Final calculated spanning tree

After the comparison processes described in Table 4, a spanning tree with Device A as the root bridge is established, as shown in Figure 5.

Figure 5 The final calculated spanning tree



The configuration BPDU forwarding mechanism of STP

The configuration BPDUs of STP are forwarded according to these guidelines:

- Upon network initiation, every device regards itself as the root bridge and generates configuration BPDUs with itself as the root. Then it sends the configuration BPDUs at a regular hello interval.
- If the root port receives a configuration BPDU superior to the configuration BPDU of the port, the device performs the following operations:
  - Increases the message age carried in the configuration BPDU.
  - Starts a timer to time the configuration BPDU.
  - Sends this configuration BPDU through the designated port.
- If a designated port receives a configuration BPDU with a lower priority than its configuration BPDU, the port immediately responds with its configuration BPDU.
- If a path fails, the root port on this path no longer receives new configuration BPDUs and the old configuration BPDUs will be discarded due to timeout. The device generates a configuration BPDU with itself as the root and sends the BPDUs and TCN BPDUs. This triggers a new spanning tree calculation process to establish a new path to restore the network connectivity.

However, the newly calculated configuration BPDU cannot be propagated throughout the network immediately. As a result, the old root ports and designated ports that have not detected the topology change continue forwarding data along the old path. If the new root ports and designated ports begin to forward data as soon as they are elected, a temporary loop might occur.

STP timers

The most important timing parameters in STP calculation are forward delay, hello time, and max age.

- Forward delay

  Forward delay is the delay time for port state transition. By default, the forward delay is 15 seconds.

  A path failure can cause spanning tree re-calculation to adapt the spanning tree structure to the change. However, the resulting new configuration BPDU cannot propagate throughout the network immediately. If the newly elected root ports and designated ports start to forward data immediately, a temporary loop will likely occur.

  The newly elected root ports or designated ports must go through the listening and learning states before they transit to the forwarding state. This requires twice the forward delay time and allows the new configuration BPDU to propagate throughout the network.

- Hello time

  The device sends configuration BPDUs at the hello time interval to the neighboring devices to ensure that the paths are fault-free. By default, the hello time is 2 seconds. If the device does not receive configuration BPDUs within the timeout period, it recalculates the spanning tree. The formula for calculating the timeout period is timeout period = timeout factor × 3 × hello time.

- Max age

  The device uses the max age to determine whether a stored configuration BPDU has expired and discards it if the max age is exceeded. By default, the max age is 20 seconds. In the CIST of an MSTP network, the device uses the max age timer to determine whether a configuration BPDU received by a port has expired. If it is expired, a new spanning tree calculation process starts. The max age timer does not take effect on MSTIs.

If a port does not receive any configuration BPDUs within the timeout period, the port transits to the listening state. The device will recalculate the spanning tree. It takes the port 50 seconds to transit back to the forwarding state. This period includes 20 seconds for the max age, 15 seconds for the listening state, and 15 seconds for the learning state.

To ensure a fast topology convergence, make sure the timer settings meet the following formulas:

- 2 × (forward delay – 1 second) ≥ max age
- Max age ≥ 2 × (hello time + 1 second)

About RSTP

RSTP achieves rapid network convergence by allowing a newly elected root port or designated port to enter the forwarding state much faster than STP.

RSTP protocol frames

An RSTP BPDU uses the same format as an STP BPDU except that a Version1 length field is added to the payload of RSTP BPDUs. The differences between an RSTP BPDU and an STP BPDU are as follows:

- **Protocol version ID**—The value is 0x02 for RSTP.
- **BPDU type**—The value is 0x02 for RSTP BPDUs.
- **Flags**—All 8 bits are used.
- **Version1 length**—The value is 0x00, which means no version 1 protocol information is present.

RSTP does not use TCN BPDUs to advertise topology changes. RSTP floods BPDUs with the TC flag set in the network to advertise topology changes.

Basic concepts in RSTP

Port roles

In addition to root port and designated port, RSTP also uses the following port roles:

- **Alternate port**—Acts as the backup port for a root port. When the root port is blocked, the alternate port takes over.
- **Backup port**—Acts as the backup port of a designated port. When the designated port is invalid, the backup port becomes the new designated port. A loop occurs when two ports of the same spanning tree device are connected, so the device blocks one of the ports. The blocked port is the backup port.
- **Edge port**—Directly connects to a user host rather than a network device or network segment.

Port states

RSTP uses the discarding state to replace the disabled, blocking, and listening states in STP. Table 5 shows the differences between the port states in RSTP and STP.

Table 5 Port state differences between RSTP and STP

STP port state	RSTP port state	Sends BPDU	Learns MAC addresses	Forwards user data
Disabled	Discarding	No	No	No
Blocking	Discarding	No	No	No
Listening	Discarding	Yes	No	No
Learning	Learning	Yes	Yes	No
Forwarding	Forwarding	Yes	Yes	Yes

How RSTP works

During RSTP calculation, the following events occur:

- If a port in discarding state becomes an alternate port, it retains its state.
- If a port in discarding state is elected as the root port or designated port, it enters the learning state after the forward delay. The port learns MAC addresses, and enters the forwarding state after another forward delay.
    - A newly elected RSTP root port rapidly enters the forwarding state if the following requirements are met:
        - The old root port on the device has stopped forwarding data.
        - The upstream designated port has started forwarding data.
    - A newly elected RSTP designated port rapidly enters the forwarding state if one of the following requirements is met:
        - The designated port is configured as an edge port which directly connects to a user terminal.
        - The designated port connects to a point-to-point link and receives a handshake response from the directly connected device.

RSTP BPDU processing

In RSTP, a non-root bridge actively sends RSTP BPDUs at the hello time through designated ports without waiting for the root bridge to send RSTP BPDUs. This enables RSTP to quickly detect link failures. If a device fails to receive any RSTP BPDUs on a port within triple the hello time, the device considers that a link failure has occurred. After the stored configuration BPDU expires, the device floods RSTP BPDUs with the TC flag set to initiate a new RSTP calculation.

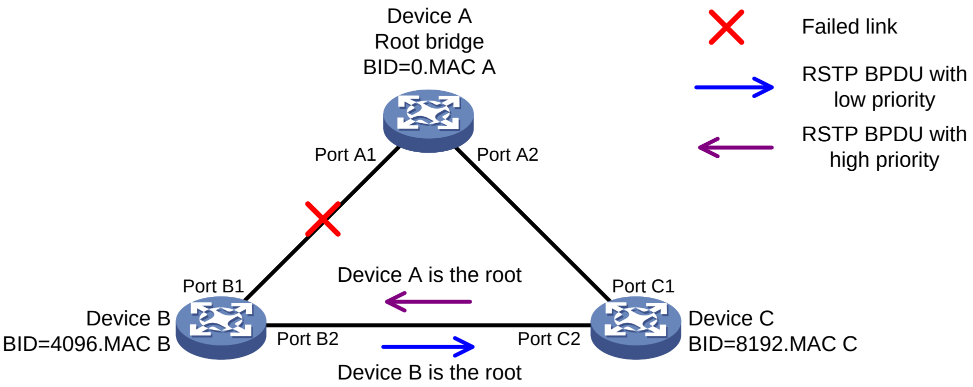
In RSTP, a port in blocking state can immediately respond to an RSTP BPDU with a lower priority than its own BPDU.

As shown in Figure 6, Device A is the root bridge. The priority of Device B is higher than the priority of Device C. Port C2 on Device C is blocked.

When the link between Device A and Device B fails, the following events occur:

1. Device B sends an RSTP BPDU with itself as the root bridge to Device C.
2. Device C compares the RSTP BPDU with its own BPDU.
3. Because the RSTP BPDU from Device B has a lower priority, Device C sends its own BPDU to Device B.
4. Device B considers that Port B2 is the root port and stops sending RSTP BPDUs to Device C.

Figure 6 BPDU processing in RSTP



About PVST

In an STP- or RSTP-enabled LAN, all bridges share one spanning tree. Traffic from all VLANs is forwarded along the spanning tree, and ports cannot be blocked on a per-VLAN basis to prune loops.

PVST allows every VLAN to have its own spanning tree, which increases usage of links and bandwidth. Because each VLAN runs RSTP independently, a spanning tree only serves its VLAN.

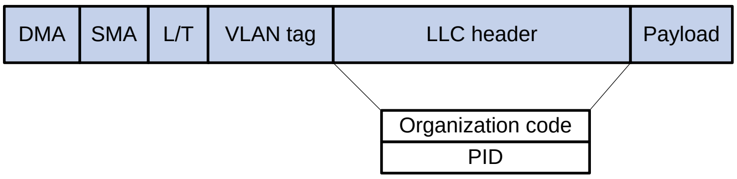
A PVST-enabled NSFOCUS device can communicate with a third-party device that is running Rapid PVST or PVST. The PVST-enabled NSFOCUS device supports fast network convergence like RSTP when connected to PVST-enabled NSFOCUS devices or third-party devices enabled with Rapid PVST.

PVST protocol frames

As shown in Figure 7, a PVST BPDU uses the same format as an RSTP BPDU except the following differences:

- The destination MAC address of a PVST BPDU is 01-00-0c-cc-cc-cd, which is a private MAC address.
- Each PVST BPDU carries a VLAN tag. The VLAN tag identifies the VLAN to which the PVST BPDU belongs.
- The organization code and PID fields are added to the LLC header of the PVST BPDU.

Figure 7 PVST BPDU format



A port's link type determines the type of BPDUs the port sends.

- An access port sends RSTP BPDUs.
- A trunk or hybrid port sends RSTP BPDUs in the default VLAN and sends PVST BPDUs in other VLANs.

How PVST works

PVST implements per-VLAN spanning tree calculation by mapping each VLAN to an MSTI. In PVST, each VLAN runs RSTP independently to maintain its own spanning tree without affecting the spanning trees of other VLANs. In this way, loops in each VLAN are eliminated and traffic of different VLANs is load shared over links. PVST uses RSTP BPDUs in the default VLAN and PVST BPDUs in other VLANs for spanning tree calculation.

PVST uses the same port roles and port states as RSTP for rapid transition. For more information, see "Basic concepts in RSTP."

About MSTP

MSTP features

Developed based on IEEE 802.1s, MSTP overcomes the limitations of STP, RSTP, and PVST. In addition to supporting rapid network convergence, it allows data flows of different VLANs to be forwarded along separate paths. This provides a better load sharing mechanism for redundant links.

MSTP provides the following features:

- MSTP divides a switched network into multiple regions, each of which contains multiple spanning trees that are independent of one another.
- MSTP supports mapping VLANs to spanning tree instances by means of a VLAN-to-instance mapping table. MSTP can reduce communication overheads and resource usage by mapping multiple VLANs to one instance.
- MSTP prunes a loop network into a loop-free tree, which avoids proliferation and endless cycling of frames in a loop network. In addition, it supports load balancing of VLAN data by providing multiple redundant paths for data forwarding.
- MSTP is compatible with STP and RSTP, and partially compatible with PVST.

MSTP protocol frames

Figure 8 shows the format of an MSTP BPDU.

Figure 8 MSTP BPDU format

Fields	Byte
Protocol ID	2
Protocol version ID	1
BPDU type	1
Flags	1
Root ID	8
Root path cost	4
Bridge ID	8
Port ID	2
Message age	2
Max age	2
Hello time	2
Forward delay	2
Version1 length=0	1
Version3 length	2
MST configuration ID	51
CIST IRPC	4
CIST bridge ID	8
CIST remaining ID	1
MSTI configuration messages	LEN

MSTP-specific fields

The first 13 fields of an MSTP BPDU are the same as an RSTP BPDU. The other six fields are unique to MSTP.

- **Protocol version ID**—The value is 0x03 for MSTP.
- **BPDU type**—The value is 0x02 for RSTP/MSTP BPDUs.
- **Root ID**—ID of the common root bridge.
- **Root path cost**—CIST external path cost.
- **Bridge ID**—ID of the regional root for the IST or an MSTI.
- **Port ID**—ID of the designated port in the CIST.
- **Version3 length**—Length of the MSTP-specific fields. Devices use this field for verification upon receiving an MSTP BPDU.
- **MST configuration ID**—Includes the format selector, configuration name, revision level, and configuration digest. The value for format selector is fixed at 0x00. The other parameters are used to identify the MST region for the originating bridge.
- **CIST IRPC**—Internal root path cost (IRPC) from the originating bridge to the root of the MST region.
- **CIST bridge ID**—ID of the bridge that sends the MSTP BPDU.
- **CIST remaining ID**—Remaining hop count. This field limits the scale of the MST region. The regional root sends a BPDU with the remaining hop count set to the maximum value. Each device that receives the BPDU decrements the hop count by one. When the hop count reaches zero, the BPDU is discarded. Devices beyond the maximum hops of the MST region cannot participate in spanning tree calculation. The default remaining hop count is 20.
- **MSTI configuration messages**—Contains MSTI configuration messages. Each MSTI configuration message is 16 bytes. This field can contain 0 to 64 MSTI configuration messages. The number of the MSTI configuration messages is determined by the number of MSTIs in the MST region.

Basic concepts in MSTP

Figure 9 shows a switched network that contains four MST regions, each MST region containing four MSTP devices. Figure 10 shows the networking topology of MST region 3.

Figure 9 Basic concepts in MSTP

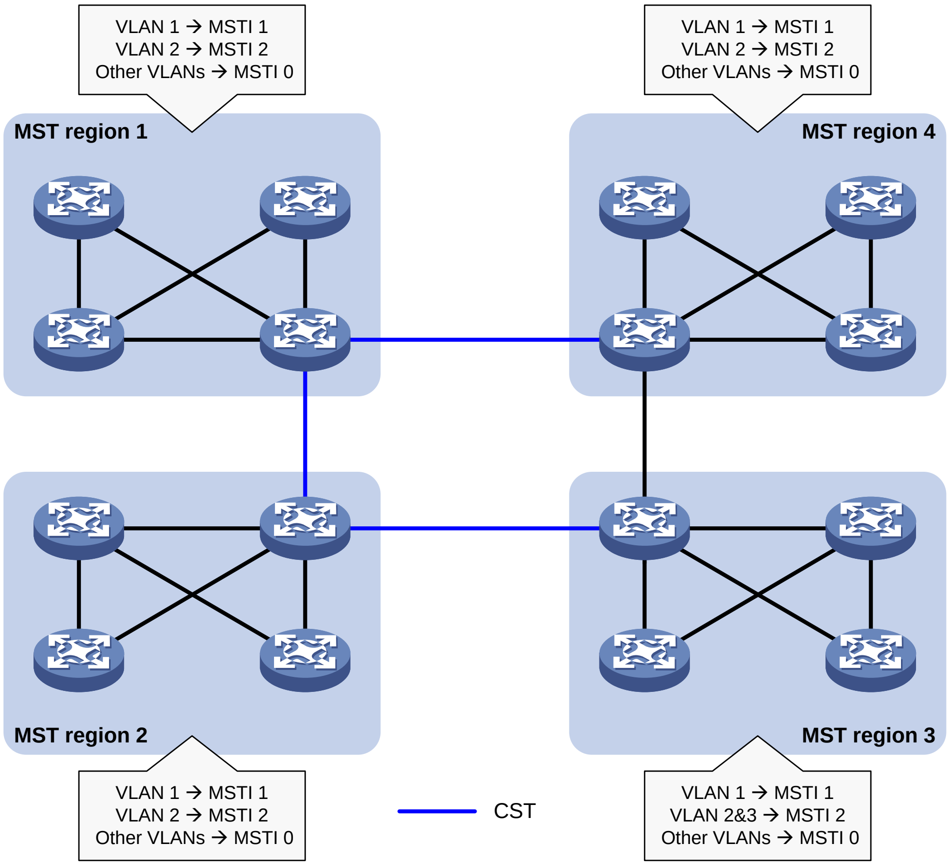
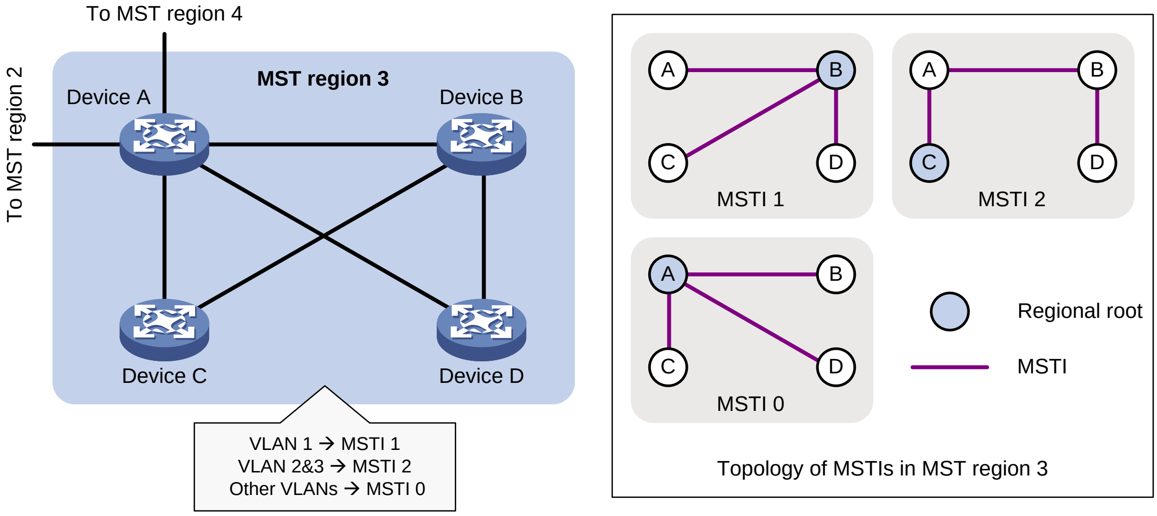


Figure 10 Network diagram and topology of MST region 3



MST region

A multiple spanning tree region (MST region) consists of multiple devices in a switched network and the network segments among them. All these devices have the following characteristics:

- A spanning tree protocol enabled
- Same region name
- Same VLAN-to-instance mapping configuration
- Same MSTP revision level
- Physically linked together

Multiple MST regions can exist in a switched network. You can assign multiple devices to the same MST region, as shown in Figure 9.

- The switched network contains four MST regions, MST region 1 through MST region 4.
- All devices in each MST region have the same MST region configuration.

MSTI

MSTP can generate multiple independent spanning trees in an MST region, and each spanning tree is mapped to the specific VLANs. Each spanning tree is referred to as a multiple spanning tree instance (MSTI).

In Figure 10, MST region 3 contains three MSTIs, MSTI 1, MSTI 2, and MSTI 0.

VLAN-to-instance mapping table

As an attribute of an MST region, the VLAN-to-instance mapping table describes the mapping relationships between VLANs and MSTIs.

In Figure 10, the VLAN-to-instance mapping table of MST region 3 is as follows:

- VLAN 1 to MSTI 1.
- VLAN 2 and VLAN 3 to MSTI 2.
- Other VLANs to MSTI 0.

MSTP achieves load balancing by means of the VLAN-to-instance mapping table.

CST

The common spanning tree (CST) is a single spanning tree that connects all MST regions in a switched network. If you regard each MST region as a device, the CST is a spanning tree calculated by these devices through STP or RSTP.

The blue lines in Figure 9 represent the CST.

IST

An internal spanning tree (IST) is a spanning tree that runs in an MST region. It is also called MSTI 0, a special MSTI to which all VLANs are mapped by default.

In Figure 9, MSTI 0 is the IST in MST region 3.

CIST

The common and internal spanning tree (CIST) is a single spanning tree that connects all devices in a switched network. It consists of the ISTs in all MST regions and the CST.

In Figure 9, the ISTs (MSTI 0) in all MST regions plus the inter-region CST constitute the CIST of the entire network.

Regional root

The root bridge of the IST or an MSTI within an MST region is the regional root of the IST or MSTI. Based on the topology, different spanning trees in an MST region might have different regional roots, as shown in MST region 3 in Figure 10.

- The regional root of MSTI 1 is Device B.
- The regional root of MSTI 2 is Device C.
- The regional root of MSTI 0 (also known as the IST) is Device A.

Common root bridge

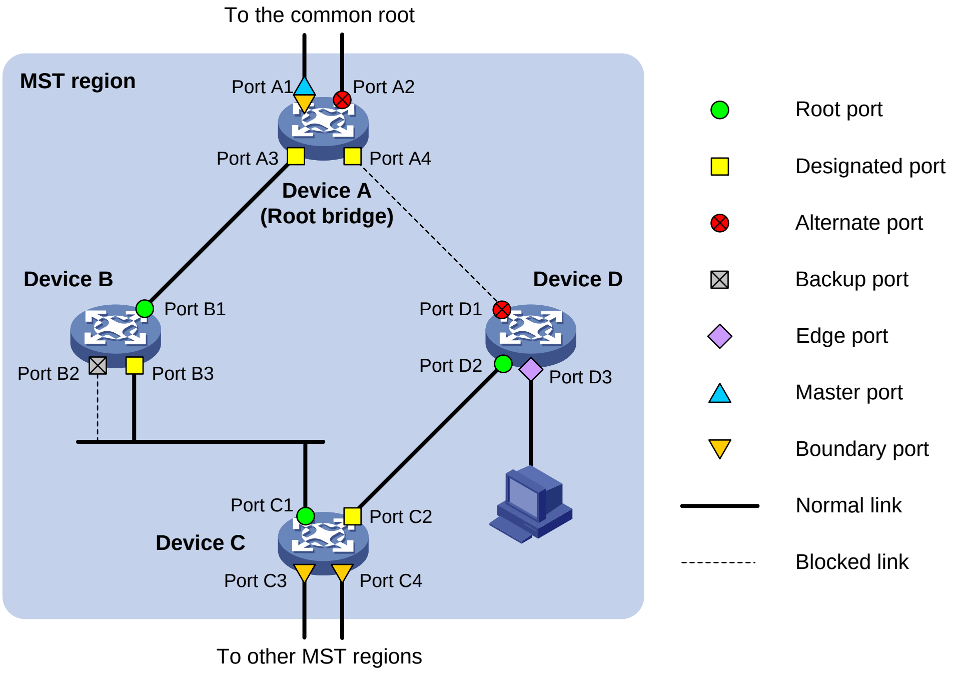
The common root bridge is the root bridge of the CIST.

In Figure 9, the common root bridge is a device in MST region 1.

Port roles

A port can play different roles in different MSTIs. As shown in Figure 11, an MST region contains Device A, Device B, Device C, and Device D. Port A1 and port A2 of Device A connect to the common root bridge. Port B2 and Port B3 of Device B form a loop. Port C3 and Port C4 of Device C connect to other MST regions. Port D3 of Device D directly connects to a host.

Figure 11 Port roles



MSTP calculation involves the following port roles:

- **Root port**—Forwards data for a non-root bridge to the root bridge. The root bridge does not have any root port.
- **Designated port**—Forwards data to the downstream network segment or device.
- **Alternate port**—Acts as the backup port for a root port or master port. When the root port or master port is blocked, the alternate port takes over.
- **Backup port**—Acts as the backup port of a designated port. When the designated port is invalid, the backup port becomes the new designated port. A loop occurs when two ports of the same spanning tree device are connected, so the device blocks one of the ports. The blocked port acts as the backup.
- **Edge port**—Directly connects to a user host rather than a network device or network segment.
- **Master port**—Acts as a port on the shortest path from the local MST region to the common root bridge. The master port is not always located on the regional root. It is a root port on the IST or CIST and still a master port on the other MSTIs.
- **Boundary port**—Connects an MST region to another MST region or to an STP/RSTP-running device. In MSTP calculation, a boundary port's role on an MSTI is consistent with its role on the

CIST. However, that is not true with master ports. A master port on MSTIs is a root port on the CIST.

Port states

In MSTP, a port can be in one of the following states:

- **Forwarding**—The port receives and sends BPDUs, learns MAC addresses, and forwards user traffic.
- **Learning**—The port receives and sends BPDUs, learns MAC addresses, but does not forward user traffic. Learning is an intermediate port state.
- **Discarding**—The port receives and sends BPDUs, but does not learn MAC addresses or forward user traffic.

NOTE:

When in different MSTIs, a port can be in different states.

A port state is not exclusively associated with a port role. Table 6 lists the port states that each port role supports. (A check mark [√] indicates that the port supports this state, while a dash [—] indicates that the port does not support this state.)

Table 6 Port states that different port roles support

Port role (right) Port state (below)	Root port/master port	Designated port	Alternate port	Backup port
Forwarding	√	√	—	—
Learning	√	√	—	—
Discarding	√	√	√	√

How MSTP works

MSTP divides an entire Layer 2 network into multiple MST regions, which are connected by a calculated CST. Inside an MST region, multiple spanning trees, called MSTIs, are calculated. Among these MSTIs, MSTI 0 is the IST.

Like STP, MSTP uses configuration BPDUs to calculate spanning trees. An important difference is that an MSTP BPDU carries the MSTP configuration of the bridge from which the BPDU is sent.

CIST calculation

During the CIST calculation, the following process takes place:

- The device with the highest priority is elected as the root bridge of the CIST.
- MSTP generates an IST within each MST region through calculation.
- MSTP regards each MST region as a single device and generates a CST among these MST regions through calculation.

The CST and ISTs constitute the CIST of the entire network.

MSTI calculation

Within an MST region, MSTP generates different MSTIs for different VLANs based on the VLAN-to-instance mappings. For each spanning tree, MSTP performs a separate calculation process similar to spanning tree calculation in STP. For more information, see "Calculation process of the STP algorithm."

In MSTP, a VLAN frame is forwarded along the following paths:

- Within an MST region, the frame is forwarded along the corresponding MSTI.
- Between two MST regions, the frame is forwarded along the CST.

MSTP implementation on devices

MSTP is compatible with STP and RSTP. Devices that are running MSTP and that are used for spanning tree calculation can identify STP and RSTP protocol frames.

In addition to basic MSTP features, the following features are provided for ease of management:

- Root bridge hold.
- Root bridge backup.
- Root guard.
- BPDU guard.
- Loop guard.
- TC-BPDU guard.
- Port role restriction.
- TC-BPDU transmission restriction.

Rapid transition mechanism

In STP, a port must wait twice the forward delay (30 seconds by default) before it transits from the blocking state to the forwarding state. The forward delay is related to the hello time and network diameter. If the forward delay is too short, loops might occur. This affects the stability of the network.

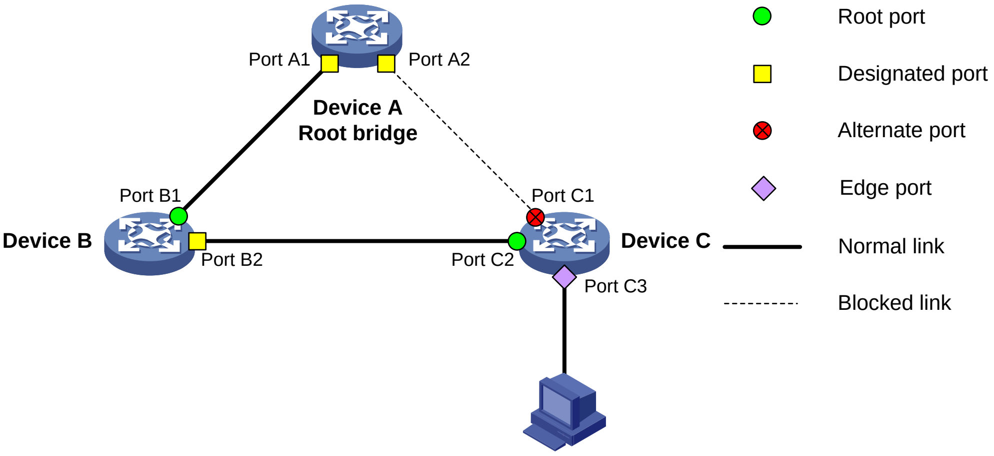
RSTP, PVST, and MSTP all use the rapid transition mechanism to speed up port state transition for edge ports, root ports, and designated ports. The rapid transition mechanism for designated ports is also known as the proposal/agreement (P/A)_transition.

Edge port rapid transition

As shown in Figure 12, Port C3 is an edge port connected to a host. When a network topology change occurs, the port can immediately transit from the blocking state to the forwarding state because no loop will be caused.

Because a device cannot determine whether a port is directly connected to a terminal, you must manually configure the port as an edge port.

Figure 12 Edge port rapid transition

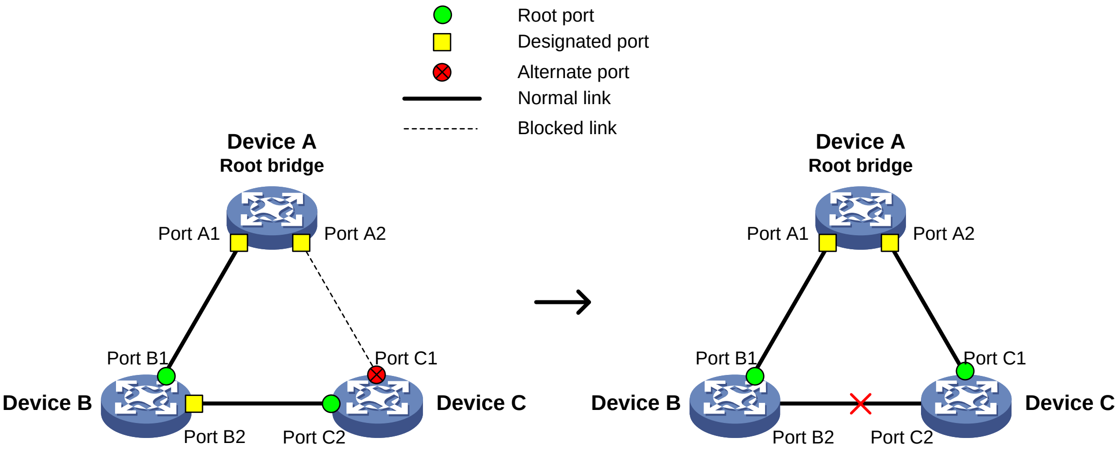


Root port rapid transition

When a root port is blocked, the bridge will elect the alternate port with the highest priority as the new root port. If the new root port's peer is in the forwarding state, the new root port immediately transits to the forwarding state.

As shown in Figure 13, Port C2 on Device C is a root port and Port C1 is an alternate port. When Port C2 transits to the blocking state, Port C1 is elected as the root port and immediately transits to the forwarding state.

Figure 13 Root port rapid transition



P/A transition

The P/A transition enables a designated port to rapidly transit to the forwarding state after a handshake with its peer. The P/A transition applies only to point-to-point links.

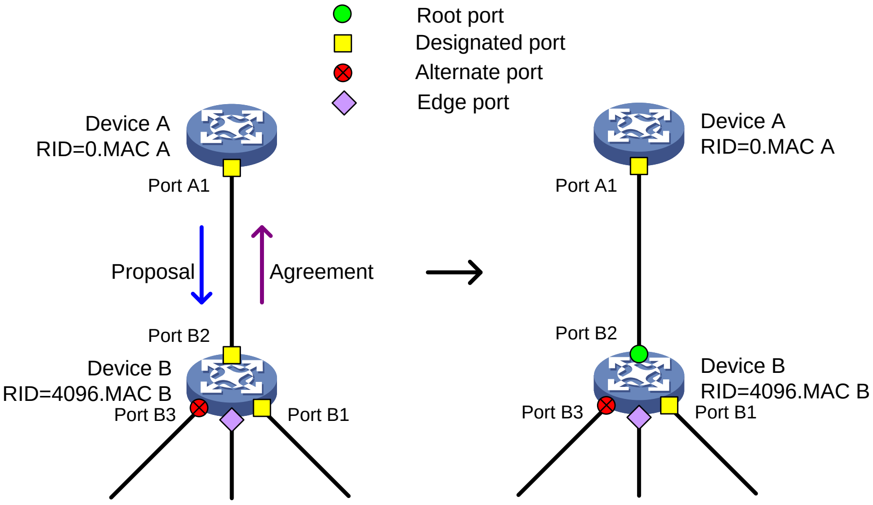
P/A transition for RSTP and PVST

In RSTP or PVST, the ports on a new link or recovered link are designated ports in blocking state. When one of the designated ports transits to the discarding or learning state, it sets the proposal flag in its BPDU. Its peer bridge receives the BPDU and determines whether the receiving port is the root port. If it is the root port, the bridge blocks the other ports except edge ports. The bridge then replies an agreement BPDU to the designated port. The designated port immediately transits to the forwarding state upon receiving the agreement BPDU. If the designated port does not receive the agreement BPDU, it waits for twice the forward delay to transit to the forwarding state.

As shown in Figure 14, the P/A transition operates as follows:

1. Device A sends a proposal BPDU to Device B through Port A1.
2. Device B receives the proposal BPDU on Port B2. Port B2 is elected as the root port.
3. Device B blocks its designated port Port B1 and alternate port Port B3 to eliminate loops.
4. The root port Port B2 transits to the forwarding state and sends an agreement BPDU to Device A.
5. The designated port Port A1 on Device A immediately transits to the forwarding state after receiving the agreement BPDU.

Figure 14 P/A transition for RSTP and PVST



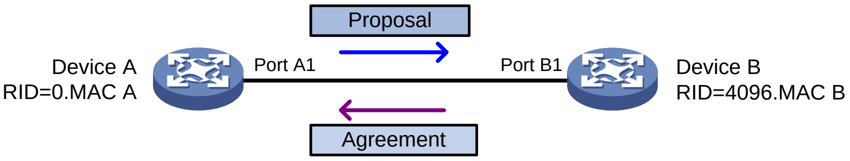
P/A transition for MSTP.

In MSTP, an upstream bridge sets both the proposal and agreement flags in its BPDU. If a downstream bridge receives the BPDU and its receiving port is elected as the root port, the bridge blocks all the other ports except edge ports. The downstream bridge then replies an agreement BPDU to the upstream bridge. The upstream port immediately transits to the forwarding state upon receiving the agreement BPDU. If the upstream port does not receive the agreement BPDU, it waits for twice the forward delay to transit to the forwarding state.

As shown in Figure 15, the P/A transition operates as follows:

1. Device A sets the proposal and agreement flags in its BPDU and sends it to Device B through Port A1.
2. Device B receives the BPDU. Port B1 of Device B is elected as the root port.
3. Device B then blocks all its ports except the edge ports.
4. The root port Port B1 of Device B transits to the forwarding state and sends an agreement BPDU to Device A.
5. Port A1 of Device A immediately transits to the forwarding state upon receiving the agreement BPDU.

Figure 15 P/A transition for MSTP



Protocols and standards

MSTP is documented in the following protocols and standards:

- IEEE 802.1d, *Media Access Control (MAC) Bridges*
- IEEE 802.1w, *Part 3: Media Access Control (MAC) Bridges—Amendment 2: Rapid Reconfiguration*
- IEEE 802.1s, *Virtual Bridged Local Area Networks—Amendment 3: Multiple Spanning Trees*

- IEEE 802.1Q-REV/D1.3, *Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks —Clause 13: Spanning tree Protocols*

Configuring spanning tree protocols

Restrictions and guidelines: spanning tree protocol configuration

Restrictions: Interface configuration

- Some spanning tree features are supported in Layer 2 Ethernet interface view and Layer 2 aggregate interface view. Unless otherwise stated, these views are collectively referred to as interface view in this document. .
- Configurations made in system view take effect globally. Configurations made in Layer 2 Ethernet interface view take effect only on the interface. Configurations made in Layer 2 aggregate interface view take effect only on the aggregate interface. Configurations made on an aggregation member port can take effect only after the port is removed from the aggregation group.
- After you enable a spanning tree protocol on a Layer 2 aggregate interface, the system performs spanning tree calculation on the Layer 2 aggregate interface. It does not perform spanning tree calculation on the aggregation member ports. The spanning tree protocol enable state and forwarding state of each selected member port are consistent with those of the corresponding Layer 2 aggregate interface.
- The member ports of an aggregation group do not participate in spanning tree calculation. However, the ports still reserve their spanning tree configurations for participating in spanning tree calculation after leaving the aggregation group.

Spanning tree protocol tasks at a glance

STP tasks at a glance

Configuring the root bridge

To configure the root bridge in STP mode, perform the following tasks:

1. Setting the spanning tree mode

   Set the spanning tree mode to STP.
2. (Optional.) Configuring the root bridge or a secondary root bridge
3. (Optional.) Configuring the device priority
4. (Optional.) Configuring parameters that affects STP topology convergence
   - Configuring the network diameter of a switched network
   - Setting spanning tree timers
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
5. (Optional.) Enabling outputting port state transition information
6. Enabling the spanning tree feature
7. (Optional.) Configuring advanced spanning tree features
   - Configuring TC Snooping

  - Configuring protection features
  - Enabling SNMP notifications for new-root election and topology change events

Configuring the leaf nodes

To configure the leaf nodes in STP mode, perform the following tasks:

1. Setting the spanning tree mode
   Set the spanning tree mode to STP.
2. (Optional.) Configuring the device priority
3. (Optional.) Configuring parameters that affects STP topology convergence
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
   - Configuring path costs of ports
   - Configuring the port priority
4. (Optional.) Enabling outputting port state transition information
5. Enabling the spanning tree feature
6. (Optional.) Configuring advanced spanning tree features
   - Configuring TC Snooping
   - Configuring protection features
   - Enabling SNMP notifications for new-root election and topology change events

RSTP tasks at a glance

Configuring the root bridge

To configure the root bridge in RSTP mode, perform the following tasks:

1. Setting the spanning tree mode
   Set the spanning tree mode to RSTP.
2. (Optional.) Configuring the root bridge or a secondary root bridge
3. (Optional.) Configuring the device priority
4. (Optional.) Configuring parameters that affects RSTP topology convergence
   - Configuring the network diameter of a switched network
   - Setting spanning tree timers
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
   - Configuring edge ports
   - Configuring the port link type
5. (Optional.) Enabling outputting port state transition information
6. Enabling the spanning tree feature
7. (Optional.) Configuring advanced spanning tree features
   - Performing mCheck
   - Configuring TC Snooping
   - Configuring protection features
   - Enabling SNMP notifications for new-root election and topology change events

Configuring the leaf nodes

To configure the leaf nodes in RSTP mode, perform the following tasks:

1. Setting the spanning tree mode

   Set the spanning tree mode to RSTP.
2. (Optional.) Configuring the device priority
3. (Optional.) Configuring parameters that affects RSTP topology convergence
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
   - Configuring edge ports
   - Configuring path costs of ports
   - Configuring the port priority
   - Configuring the port link type
4. (Optional.) Enabling outputting port state transition information
5. Enabling the spanning tree feature
6. (Optional.) Configuring advanced spanning tree features
   - Performing mCheck
   - Configuring TC Snooping
   - Configuring protection features
   - Enabling SNMP notifications for new-root election and topology change events

PVST tasks at a glance

Configuring the root bridge

To configure the root bridge in PVST mode, perform the following tasks:

1. Setting the spanning tree mode

   Set the spanning tree mode to PVST.
2. (Optional.) Configuring the root bridge or a secondary root bridge
3. (Optional.) Configuring the device priority
4. (Optional.) Configuring parameters that affects PVST topology convergence
   - Configuring the network diameter of a switched network
   - Setting spanning tree timers
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
   - Configuring edge ports
   - Configuring the port link type
5. (Optional.) Enabling outputting port state transition information
6. Enabling the spanning tree feature
7. (Optional.) Configuring advanced spanning tree features
   - Performing mCheck
   - Disabling inconsistent PVID protection
   - Configuring protection features
   - Enabling SNMP notifications for new-root election and topology change events

Configuring the leaf nodes

To configure the leaf nodes in PVST mode, perform the following tasks:

1. Setting the spanning tree mode

   Set the spanning tree mode to PVST.

2. (Optional.) Configuring the device priority
3. (Optional.) Configuring parameters that affects PVST topology convergence
    - Setting the timeout factor
    - Configuring the BPDU transmission rate
    - Configuring edge ports
    - Configuring path costs of ports
    - Configuring the port priority
    - Configuring the port link type
4. (Optional.) Enabling outputting port state transition information
5. Enabling the spanning tree feature
6. (Optional.) Configuring advanced spanning tree features
    - Performing mCheck
    - Disabling inconsistent PVID protection
    - Configuring protection features
    - Enabling SNMP notifications for new-root election and topology change events

MSTP tasks at a glance

Configuring the root bridge

To configure the root bridge in MSTP mode, perform the following tasks:

1. Setting the spanning tree mode

    Set the spanning tree mode to MSTP.
2. Configuring an MST region
3. (Optional.) Configuring the root bridge or a secondary root bridge
4. (Optional.) Configuring the device priority
5. (Optional.) Configuring parameters that affects MSTP topology convergence
    - Configuring the maximum hops of an MST region
    - Configuring the network diameter of a switched network
    - Setting spanning tree timers
    - Setting the timeout factor
    - Configuring the BPDU transmission rate
    - Configuring edge ports
    - Configuring the port link type
6. (Optional.) Configuring the mode a port uses to recognize and send MSTP frames
7. (Optional.) Enabling outputting port state transition information
8. Enabling the spanning tree feature
9. (Optional.) Configuring advanced spanning tree features
    - Performing mCheck
    - Configuring Digest Snooping
    - Configuring No Agreement Check
    - Configuring TC Snooping
    - Configuring protection features
    - Enabling SNMP notifications for new-root election and topology change events

Configuring the leaf nodes

To configure the leaf nodes in MSTP mode, perform the following tasks:

1. Setting the spanning tree mode
   Set the spanning tree mode to MSTP.
2. Configuring an MST region
3. (Optional.) Configuring the device priority
4. (Optional.) Configuring parameters that affects MSTP topology convergence
   - Setting the timeout factor
   - Configuring the BPDU transmission rate
   - Configuring edge ports
   - Configuring path costs of ports
   - Configuring the port priority
   - Configuring the port link type
5. (Optional.) Configuring the mode a port uses to recognize and send MSTP frames
6. (Optional.) Enabling outputting port state transition information
7. Enabling the spanning tree feature
8. (Optional.) Configuring advanced spanning tree features
   - Performing mCheck
   - Configuring Digest Snooping
   - Configuring No Agreement Check
   - Configuring TC Snooping
   - Configuring protection features
   - Enabling SNMP notifications for new-root election and topology change events

Setting the spanning tree mode

About this task

The spanning tree modes include:

- **STP mode**—All ports of the device send STP BPDUs. Select this mode when the peer device of a port supports only STP.
- **RSTP mode**—All ports of the device send RSTP BPDUs. A port in this mode automatically transits to the STP mode when it receives STP BPDUs from the peer device. A port in this mode does not transit to the MSTP mode when it receives MSTP BPDUs from the peer device.
- **PVST mode**—All ports of the device send PVST BPDUs. Each VLAN maintains a spanning tree. In a network, the amount of spanning trees maintained by all devices equals the number of PVST-enabled VLANs multiplied by the number of PVST-enabled ports. If the amount of spanning trees exceeds the capacity of the network, device CPUs will be overloaded. Packet forwarding is interrupted, and the network becomes unstable.
- **MSTP mode**—All ports of the device send MSTP BPDUs. A port in this mode automatically transits to the STP mode when receiving STP BPDUs from the peer device. A port in this mode does not transit to the RSTP mode when receiving RSTP BPDUs from the peer device.

Restrictions and guidelines

The MSTP mode is compatible with the RSTP mode, and the RSTP mode is compatible with the STP mode.

Compatibility of the PVST mode depends on the link type of a port.

- On an access port, the PVST mode is compatible with other spanning tree modes in all VLANs.
- On a trunk port or hybrid port, the PVST mode is compatible with other spanning tree modes only in the default VLAN.

Procedure

1. Enter system view.

   system-view
2. Set the spanning tree mode.

   stp mode { **mstp** | **pvst** | **rstp** | **stp** }

   The default setting is the MSTP mode.

Configuring an MST region

About this task

Spanning tree devices belong to the same MST region if they are both connected through a physical link and configured with the following details:

- Format selector (0 by default, not configurable).
- MST region name.
- MST region revision level.
- VLAN-to-instance mapping entries in the MST region.

The configuration of MST region-related parameters (especially the VLAN-to-instance mapping table) might cause MSTP to begin a new spanning tree calculation. To reduce the possibility of topology instability, the MST region configuration takes effect only after you activate it by doing one of the following:

- Use the **active region-configuration** command.
- Enable a spanning tree protocol by using the **stp global enable** command if the spanning tree protocol is disabled.

Restrictions and guidelines

In STP, PVST, or RSTP mode, MST region configurations do not take effect.

Procedure

1. Enter system view.

   system-view
2. Enter MST region view.

   stp region-configuration
3. Configure the MST region name.

   region-name *name*

   The default setting is the MAC address.
4. Configure the VLAN-to-instance mapping table. Choose one option as needed:
   - Map a list of VLANs to an MSTI.

     instance *instance-id* **vlan** *vlan-id-list*
   - Quickly create a VLAN-to-instance mapping table.

     vlan-mapping modulo *modulo*

   By default, all VLANs in an MST region are mapped to the CIST (or MSTI 0).
5. Configure the MSTP revision level of the MST region.

revision-level *level*

The default setting is 0.

6. (Optional.) Display the MST region configurations that are not activated yet.

   check region-configuration

7. Manually activate MST region configuration.

   active region-configuration

Configuring the root bridge or a secondary root bridge

Restrictions and guidelines

You can have the spanning tree protocol determine the root bridge of a spanning tree through calculation. You can also specify a device as the root bridge or as a secondary root bridge.

When you specify a device as the root bridge or as a secondary root bridge, follow these restrictions and guidelines:

- A device has independent roles in different spanning trees. It can act as the root bridge in one spanning tree and as a secondary root bridge in another. However, one device cannot be the root bridge and a secondary root bridge in the same spanning tree.
- If you specify the root bridge for a spanning tree, no new root bridge is elected according to the device priority settings. Once you specify a device as the root bridge or a secondary root bridge, you cannot change the priority of the device.
- You can configure a device as the root bridge by setting the device priority to 0. For the device priority configuration, see "Configuring the device priority."

Configuring the device as the root bridge of a spanning tree

1. Enter system view.

   system-view

2. Configure the device as the root bridge.
   - In STP/RSTP mode:

     stp root primary
   - In PVST mode:

     stp vlan *vlan-id-list* **root primary**
   - In MSTP mode:

     stp [**instance** *instance-list*] **root primary**

   By default, the device is not a root bridge.

Configuring the device as a secondary root bridge of a spanning tree

1. Enter system view.

   system-view

2. Configure the device as a secondary root bridge.
   - In STP/RSTP mode:

stp root secondary

- In PVST mode:

  stp vlan *vlan-id-list* **root secondary**
- In MSTP mode:

  stp [**instance** *instance-list*] **root secondary**

By default, the device is not a secondary root bridge.

Configuring the device priority

About this task

Device priority is a factor in calculating the spanning tree. The priority of a device determines whether the device can be elected as the root bridge of a spanning tree. A lower value indicates a higher priority. You can set the priority of a device to a low value to specify the device as the root bridge of the spanning tree. A spanning tree device can have different priorities in different spanning trees.

During root bridge selection, if all devices in a spanning tree have the same priority, the one with the lowest MAC address is selected. You cannot change the priority of a device after it is configured as the root bridge or as a secondary root bridge.

Procedure

1. Enter system view.

   system-view
2. Configure the priority of the device.
   - In STP/RSTP mode:

     stp priority *priority*
   - In PVST mode:

     stp vlan *vlan-id-list* **priority** *priority*
   - In MSTP mode:

     stp [**instance** *instance-list*] **priority** *priority*

   The default setting is 32768.

Configuring the maximum hops of an MST region

About this task

Restrict the region size by setting the maximum hops of an MST region. The hop limit configured on the regional root bridge is used as the hop limit for the MST region.

Configuration BPDUs sent by the regional root bridge always have a hop count set to the maximum value. When a device receives this configuration BPDU, it decrements the hop count by one, and uses the new hop count in the BPDUs that it propagates. When the hop count of a BPDU reaches zero, it is discarded by the device that received it. Devices beyond the reach of the maximum hops can no longer participate in spanning tree calculations, so the size of the MST region is limited.

Restrictions and guidelines

Make this configuration only on the root bridge. All other devices in the MST region use the maximum hop value set for the root bridge.

You can configure the maximum hops of an MST region based on the STP network size. As a best practice, set the maximum hops to a value that is greater than the maximum hops of each edge device to the root bridge.

Procedure

1. Enter system view.

   system-view

2. Configure the maximum hops of the MST region.

   stp max-hops *hops*

   The default setting is 20.

Configuring the network diameter of a switched network

About this task

Any two terminal devices in a switched network can reach each other through a specific path, and there are a series of devices on the path. The switched network diameter is the maximum number of devices on the path for an edge device to reach another one in the switched network through the root bridge. The network diameter indicates the network size. The bigger the diameter, the larger the network size.

Based on the network diameter you configured, the system automatically sets an optimal hello time, forward delay, and max age for the device.

In STP, RSTP, or MSTP mode, each MST region is considered a device. The configured network diameter takes effect only on the CIST (or the common root bridge) but not on other MSTIs.

In PVST mode, the configured network diameter takes effect only on the root bridges of the specified VLANs.

Procedure

1. Enter system view.

   system-view

2. Configure the network diameter of the switched network.

   - In STP/RSTP/MSTP mode:

     stp bridge-diameter *diameter*

   - In PVST mode:

     stp vlan *vlan-id-list* **bridge-diameter** *diameter*

   The default setting is 7.

Setting spanning tree timers

About this task

The following timers are used for spanning tree calculation:

- **Forward delay**—Delay time for port state transition. To prevent temporary loops on a network, the spanning tree feature sets an intermediate port state (the learning state) before it transits from the discarding state to the forwarding state. The feature also requires that the port transit its state after a forward delay timer. This ensures that the state transition of the local port stays synchronized with the peer.
- **Hello time**—Interval at which the device sends configuration BPDUs to detect link failures. If the device does not receive configuration BPDUs within the timeout period, it recalculates the spanning tree. The formula for calculating the timeout period is timeout period = timeout factor × 3 × hello time.

- **Max age**—In the CIST of an MSTP network, the device uses the max age timer to determine whether a configuration BPDU received by a port has expired. If it is expired, a new spanning tree calculation process starts. The max age timer does not take effect on MSTIs.

To ensure a fast topology convergence, make sure the timer settings meet the following formulas:

- 2 × (forward delay – 1 second) ≥ max age
- Max age ≥ 2 × (hello time + 1 second)

As a best practice, specify the network diameter and letting spanning tree protocols automatically calculate the timers based on the network diameter instead of manually setting the spanning tree timers. If the network diameter uses the default value, the timers also use their default values.

Set the timers only on the root bridge. The timer settings on the root bridge apply to all devices on the entire switched network.

Restrictions and guidelines

- The length of the forward delay is related to the network diameter of the switched network. The larger the network diameter is, the longer the forward delay time should be. As a best practice, use the automatically calculated value because inappropriate forward delay setting might cause temporary redundant paths or increase the network convergence time.
- An appropriate hello time setting enables the device to promptly detect link failures on the network without using excessive network resources. If the hello time is too long, the device mistakes packet loss for a link failure and triggers a new spanning tree calculation process. If the hello time is too short, the device frequently sends the same configuration BPDUs, which wastes device and network resources. As a best practice, use the automatically calculated value.
- If the max age timer is too short, the device frequently begins spanning tree calculations and might mistake network congestion as a link failure. If the max age timer is too long, the device might fail to promptly detect link failures and quickly launch spanning tree calculations, reducing the auto-sensing capability of the network. As a best practice, use the automatically calculated value.

Procedure

1. Enter system view.

   `system-view`

2. Set the forward delay timer.
   - In STP/RSTP/MSTP mode:

     stp timer forward-delay *time*
   - In PVST mode:

     stp vlan *vlan-id-list* **timer forward-delay** *time*

   The default setting is 15 seconds.

3. Set the hello timer.
   - In STP/RSTP/MSTP mode:

     stp timer hello *time*
   - In PVST mode:

     stp vlan *vlan-id-list* **timer hello** *time*

   The default setting is 2 seconds.

4. Set the max age timer.
   - In STP/RSTP/MSTP mode:

     stp timer max-age *time*
   - In PVST mode:

     stp vlan *vlan-id-list* **timer max-age** *time*

The default setting is 20 seconds.

Setting the timeout factor

About this task

The timeout factor is a parameter used to decide the timeout period. The formula for calculating the timeout period is: *timeout period* = *timeout factor* × 3 × *hello time*.

In a stable network, each non-root-bridge device forwards configuration BPDUs to the downstream devices at the hello time interval to detect link failures. If a device does not receive a BPDU from the upstream device within nine times the hello time, it assumes that the upstream device has failed. Then, it starts a new spanning tree calculation process.

Restrictions and guidelines

As a best practice, set the timeout factor to 5, 6, or 7 in the following situations:

- To prevent undesired spanning tree calculations. An upstream device might be too busy to forward configuration BPDUs in time, for example, many Layer 2 interfaces are configured on the upstream device. In this case, the downstream device fails to receive a BPDU within the timeout period and then starts an undesired spanning tree calculation.
- To save network resources on a stable network.

Procedure

1. Enter system view.

```
system-view
```

2. Set the timeout factor of the device.

```
stp timer-factor factor
```

The default setting is 3.

Configuring the BPDU transmission rate

About this task

The maximum number of BPDUs a port can send within each hello time equals the BPDU transmission rate plus the hello timer value.

The higher the BPDU transmission rate, the more BPDUs are sent within each hello time, and the more system resources are used. By setting an appropriate BPDU transmission rate, you can limit the rate at which the port sends BPDUs. Setting an appropriate rate also prevents spanning tree protocols from using excessive network resources when the network topology changes.

Restrictions and guidelines

The BPDU transmission rate depends on the physical status of the port and the network structure. As a best practice, use the default setting.

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Configure the BPDU transmission rate of the ports.

```
stp transmit-limit limit
```

The default setting is 10.

Configuring edge ports

About this task

If a port directly connects to a user terminal rather than another device or a shared LAN segment, this port is regarded as an edge port. When network topology change occurs, an edge port will not cause a temporary loop. Because a device does not determine whether a port is directly connected to a terminal, you must manually configure the port as an edge port. After that, the port can rapidly transit from the blocking state to the forwarding state.

Restrictions and guidelines

- If BPDU guard is disabled on a port configured as an edge port, the port becomes a non-edge port again if it receives a BPDU from another port. To restore the edge port, re-enable it.
- If a port directly connects to a user terminal, configure it as an edge port and enable BPDU guard for it. This enables the port to quickly transit to the forwarding state when ensuring network security.
- On a port, the loop guard feature and the edge port setting are mutually exclusive.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the port as an edge port.

   `stp edged-port`

   By default, all ports are non-edge ports.

Configuring path costs of ports

About path cost

Path cost is a parameter related to the link speed of a port. On a spanning tree device, a port can have different path costs in different MSTIs. Setting appropriate path costs allows VLAN traffic flows to be forwarded along different physical links, achieving VLAN-based load balancing.

You can have the device automatically calculate the default path cost, or you can configure the path cost for ports.

Specifying a standard for the default path cost calculation

About this task

You can specify a standard for the device to use in automatic calculation for the default path cost. The device supports the following standards:

- **dot1d-1998**—The device calculates the default path cost for ports based on IEEE 802.1d-1998.
- **dot1t**—The device calculates the default path cost for ports based on IEEE 802.1t.
- **legacy**—The device calculates the default path cost for ports based on a private standard.

Table 7 Mappings between the link speed (100M and below) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
0	N/A	65535	200000000	200000
10 Mbps	Single port	100	2000000	2000
	Aggregate interface containing two Selected ports		1000000	1800
	Aggregate interface containing three Selected ports		666666	1600
	Aggregate interface containing four Selected ports		500000	1400
100 Mbps	Single port	19	200000	200
	Aggregate interface containing two Selected ports		100000	180
	Aggregate interface containing three Selected ports		66666	160
	Aggregate interface containing four Selected ports		50000	140

Table 8 Mappings between the link speed (1000M) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
1000 Mbps	Single port	4	20000	20
	Aggregate interface containing two Selected ports		10000	18
	Aggregate interface containing three Selected ports		6666	16
	Aggregate interface containing four Selected ports		5000	14

Table 9 Mappings between the link speed (10G) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
10 Gbps	Single port	2	2000	2
	Aggregate interface		1000	1

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
	containing two Selected ports			
	Aggregate interface containing three Selected ports		666	1
	Aggregate interface containing four Selected ports		500	1

Table 10 Mappings between the link speed (20G) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
20 Gbps	Single port	1	1000	1
	Aggregate interface containing two Selected ports		500	1
	Aggregate interface containing three Selected ports		333	1
	Aggregate interface containing four Selected ports		250	1

Table 11 Mappings between the link speed (40G) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
40 Gbps	Single port	1	500	1
	Aggregate interface containing two Selected ports		250	1
	Aggregate interface containing three Selected ports		166	1
	Aggregate interface containing four Selected ports		125	1

Table 12 Mappings between the link speed (100G) and the path cost

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
100 Gbps	Single port	1	200	1

Link speed	Port type	Path cost		
		IEEE 802.1d-1998	IEEE 802.1t	Private standard
	Aggregate interface containing two Selected ports		100	1
	Aggregate interface containing three Selected ports		66	1
	Aggregate interface containing four Selected ports		50	1

Restrictions and guidelines

If you change the standard for the default path cost calculation, you restore the path costs to the default.

When the device calculates the path cost for an aggregate interface, IEEE 802.1t takes into account the number of Selected ports in its aggregation group. However, IEEE 802.1d-1998 does not take into account the number of Selected ports. The calculation formula of IEEE 802.1t is: Path cost = 200,000,000/link speed (in 100 kbps). The link speed is the sum of the link speed values of the Selected ports in the aggregation group.

IEEE 802.1d-1998 or the private standard always assigns the smallest possible value to a single port or aggregate interface with a speed exceeding 10 Gbps. The forwarding path selected based on this criterion might not be the best one. To solve this problem, perform one of the following tasks:

- Use **dot1t** as the standard for default path cost calculation.
- Manually set the path cost for the port (see "Configuring path costs of ports").

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify a standard for the default path costs calculation.

   ```
   stp pathcost-standard { dot1d-1998 | dot1t | legacy }
   ```

   The default standard is **legacy**.

Configuring path costs of ports

Restrictions and guidelines

When the path cost of a port changes, the system recalculates the port role and initiates a state transition.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Configure the path cost of the ports.

   - In STP/RSTP mode:

     ```
     stp cost cost-value
     ```

- In PVST mode:

  stp vlan *vlan-id-list* **cost** *cost-value*
- In MSTP mode:

  stp [**instance** *instance-list*] **cost** *cost-value*

By default, the system automatically calculates the path cost of each port.

Configuring the port priority

About this task

The priority of a port is a factor that determines whether the port can be elected as the root port of a device. If all other conditions are the same, the port with the highest priority is elected as the root port.

On a spanning tree device, a port can have different priorities and play different roles in different spanning trees. As a result, data of different VLANs can be propagated along different physical paths, implementing per-VLAN load balancing. You can set port priority values based on the actual networking requirements.

Restrictions and guidelines

When the priority of a port changes, the system recalculates the port role and initiates a state transition. Prepare for the network topology change before configuring the port priority.

Procedure

1. Enter system view.

   system-view
2. Enter interface view.

   interface *interface-type interface-number*
3. Configure the port priority.
   - In STP/RSTP mode:

     stp port priority *priority*
   - In PVST mode:

     stp vlan *vlan-id-list* **port priority** *priority*
   - In MSTP mode:

     stp [**instance** *instance-list*] **port priority** *priority*

   The default setting is 128 for all ports.

Configuring the port link type

About this task

A point-to-point link directly connects two devices. If two root ports or designated ports are connected over a point-to-point link, they can rapidly transit to the forwarding state after a proposal-agreement handshake process.

Restrictions and guidelines

- You can configure the link type as point-to-point for a Layer 2 aggregate interface or a port that operates in full duplex mode. As a best practice, use the default setting and let the device automatically detect the port link type.

- In PVST or MSTP mode, the **`stp point-to-point force-false`** or **`stp point-to-point force-true`** command configured on a port takes effect on all VLANs or all MSTIs.
- Before you set the link type of a port to point-to-point, make sure the port is connected to a point-to-point link. Otherwise, a temporary loop might occur.

Procedure

1. Enter system view.

`system-view`

2. Enter interface view.

`interface` *`interface-type interface-number`*

3. Configure the port link type.

`stp point-to-point { auto | force-false | force-true }`

By default, the link type is **auto** where the port automatically detects the link type.

Configuring the mode a port uses to recognize and send MSTP frames

About this task

A port can receive and send MSTP frames in the following formats:

- **dot1s**—802.1s-compliant standard format
- **legacy**—Compatible format

By default, the frame format recognition mode of a port is **auto**. The port automatically distinguishes the two MSTP frame formats, and determines the format of frames that it will send based on the recognized format.

You can configure the MSTP frame format on a port. Then, the port sends only MSTP frames of the configured format to communicate with devices that send frames of the same format.

By default, a port in **auto** mode sends 802.1s MSTP frames. When the port receives an MSTP frame of a legacy format, the port starts to send frames only of the legacy format. This prevents the port from frequently changing the format of sent frames. To configure the port to send 802.1s MSTP frames, shut down and then bring up the port.

Restrictions and guidelines

When the number of existing MSTIs exceeds 48, the port can send only 802.1s MSTP frames.

Procedure

1. Enter system view.

`system-view`

2. Enter interface view.

`interface` *`interface-type interface-number`*

3. Configure the mode that the port uses to recognize/send MSTP frames.

`stp compliance { auto | dot1s | legacy }`

The default setting is **auto**.

Enabling outputting port state transition information

About this task

In a large-scale spanning tree network, you can enable devices to output the port state transition information. Then, you can monitor the port states in real time.

Procedure

1. Enter system view.

   `system-view`

2. Enable outputting port state transition information.
   - In STP/RSTP mode:

     stp port-log instance 0
   - In PVST mode:

     stp port-log vlan *vlan-id-list*
   - In MSTP mode:

     stp port-log { **all** | **instance** *instance-list* }

   By default, outputting port state transition information is disabled.

Enabling the spanning tree feature

Restrictions and guidelines

You must enable the spanning tree feature for the device before any other spanning tree related configurations can take effect. In STP, RSTP, or MSTP mode, make sure the spanning tree feature is enabled globally and on the desired ports. In PVST mode, make sure the spanning tree feature is enabled globally, in the desired VLANs, and on the desired ports.

To exclude specific ports from spanning tree calculation and save CPU resources, disable the spanning tree feature for these ports with the **undo stp enable** command. Make sure no loops occur in the network after you disable the spanning tree feature on these ports.

Enabling the spanning tree feature in STP/RSTP/MSTP mode

1. Enter system view.

   system-view

2. Enable the spanning tree feature.

   stp global enable

   By default, the spanning tree feature is disabled globally.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable the spanning tree feature for the port.

   stp enable

   By default, the spanning tree feature is enabled on all ports.

Enabling the spanning tree feature in PVST mode

1. Enter system view.

   system-view

2. Enable the spanning tree feature.

   stp global enable

   By default, the spanning tree feature is disabled globally.

3. Enable the spanning tree feature in VLANs.

   stp vlan *vlan-id-list* **enable**

   By default, the spanning tree feature is enabled in VLANs.

4. Enter interface view.

   interface *interface-type interface-number*

5. Enable the spanning tree feature on the port.

   stp enable

   By default, the spanning tree feature is enabled on all ports.

Performing mCheck

About mCheck

The mCheck feature enables user intervention in the port state transition process.

When a port on an MSTP, PVST, or RSTP device connects to an STP device and receives STP BPDUs, the port automatically transits to the STP mode. However, the port cannot automatically transit back to the original mode when the following conditions exist:

- The peer STP device is shut down or removed.
- The port cannot detect the change.

To forcibly transit the port to operate in the original mode, you can perform an mCheck operation.

For example, Device A, Device B, and Device C are connected in sequence. Device A runs STP, Device B does not run any spanning tree protocol, and Device C runs RSTP, PVST, or MSTP. In this case, when Device C receives an STP BPDU transparently transmitted by Device B, the receiving port transits to the STP mode. If you configure Device B to run RSTP, PVST, or MSTP with Device C, you must perform mCheck operations on the ports interconnecting Device B and Device C.

Restrictions and guidelines

The mCheck operation takes effect on devices operating in MSTP, PVST, or RSTP mode.

Performing mCheck globally

1. Enter system view.

   system-view

2. Perform mCheck.

   stp global mcheck

Performing mCheck in interface view

1. Enter system view.
   `system-view`
2. Enter interface view.
   `interface` *`interface-type interface-number`*
3. Perform mCheck.
   `stp mcheck`

Disabling inconsistent PVID protection

About this task

In PVST, if two connected ports use different PVIDs, PVST calculation errors might occur. By default, inconsistent PVID protection is enabled to avoid PVST calculation errors. If PVID inconsistency is detected on a port, the system blocks the port.

Restrictions and guidelines

If different PVIDs are required on two connected ports, disable inconsistent PVID protection on the devices that host the ports. To avoid PVST calculation errors, make sure the following requirements are met:

- Make sure the VLANs on one device do not use the same ID as the PVID of its peer port (except the default VLAN) on another device.
- If the local port or its peer is a hybrid port, do not configure the local and peer ports as untagged members of the same VLAN.
- Disable inconsistent PVID protection on both the local device and the peer device.

This feature takes effect only when the device is operating in PVST mode.

Procedure

1. Enter system view.
   `system-view`
2. Disable the inconsistent PVID protection feature.
   `stp ignore-pvid-inconsistency`
   By default, the inconsistent PVID protection feature is enabled.

Configuring Digest Snooping

About this task

As defined in IEEE 802.1s, connected devices are in the same region only when they have the same MST region-related configurations, including:

- Region name.
- Revision level.
- VLAN-to-instance mappings.

A spanning tree device identifies devices in the same MST region by determining the configuration ID in BPDUs. The configuration ID includes the region name, revision level, and configuration digest. It is 16-byte long and is the result calculated through the HMAC-MD5 algorithm based on VLAN-to-instance mappings.

Because spanning tree implementations vary by vendor, the configuration digests calculated through private keys are different. The devices of different vendors in the same MST region cannot communicate with each other.

To enable communication between an NSFOCUS device and a third-party device in the same MST region, enable Digest Snooping on the NSFOCUS device port connecting them.

Restrictions and guidelines

△ **CAUTION:**

Use caution with global Digest Snooping in the following situations:

- When you modify the VLAN-to-instance mappings.
- When you restore the default MST region configuration.

If the local device has different VLAN-to-instance mappings than its neighboring devices, loops or traffic interruption will occur.

- Before you enable Digest Snooping, make sure associated devices of different vendors are connected and run spanning tree protocols.
- With Digest Snooping enabled, in-the-same-region verification does not require comparison of configuration digest. The VLAN-to-instance mappings must be the same on associated ports.
- To make Digest Snooping take effect, you must enable Digest Snooping both globally and on associated ports. As a best practice, enable Digest Snooping on all associated ports first and then enable it globally. This will make the configuration take effect on all configured ports and reduce impact on the network.
- To prevent loops, do not enable Digest Snooping on MST region edge ports.
- As a best practice, enable Digest Snooping first and then enable the spanning tree feature. To avoid traffic interruption, do not configure Digest Snooping when the network is already working well.

Prerequisites

Before configuring Digest Snooping, you need to make sure your NSFOCUS device and the third-party device both run spanning tree protocols properly.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable Digest Snooping on the interface.

   `stp config-digest-snooping`

   By default, Digest Snooping is disabled on ports.

4. Return to system view.

   `quit`

5. Enable Digest Snooping globally.

   `stp global config-digest-snooping`

   By default, Digest Snooping is disabled globally.

Configuring No Agreement Check

About this task

In RSTP and MSTP, the following types of messages are used for rapid state transition on designated ports:

- **Proposal**—Sent by designated ports to request rapid transition
- **Agreement**—Used to acknowledge rapid transition requests

Both RSTP and MSTP devices can perform rapid transition on a designated port only when the port receives an agreement packet from the downstream device. RSTP and MSTP devices have the following differences:

- For MSTP, the root port of the downstream device sends an agreement packet only after it receives an agreement packet from the upstream device.
- For RSTP, the downstream device sends an agreement packet whether or not an agreement packet from the upstream device is received.

Figure 16 Rapid state transition of an MSTP designated port

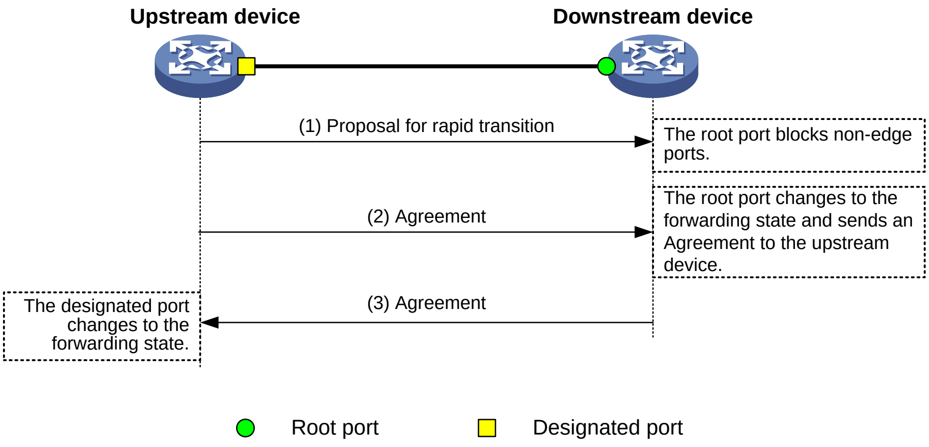
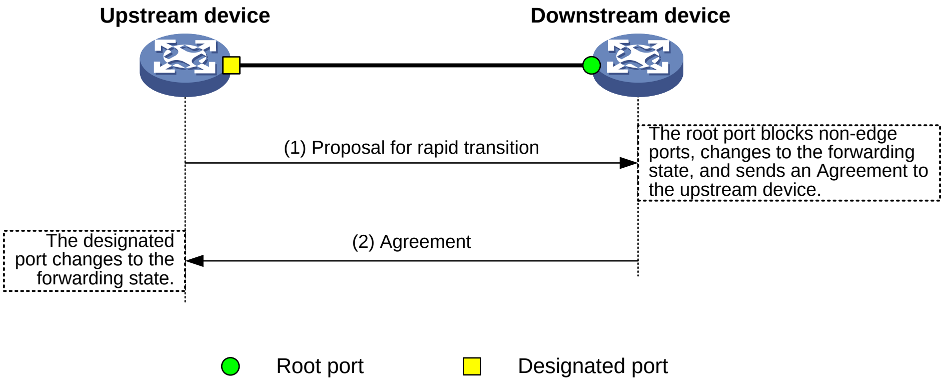

Upstream device
Downstream device
(1) Proposal for rapid transition
The root port blocks non-edge ports.
(2) Agreement
The root port changes to the forwarding state and sends an Agreement to the upstream device.
The designated port changes to the forwarding state.
(3) Agreement
Root port
Designated port

Figure 17 Rapid state transition of an RSTP designated port


Upstream device
Downstream device
(1) Proposal for rapid transition
The root port blocks non-edge ports, changes to the forwarding state, and sends an Agreement to the upstream device.
The designated port changes to the forwarding state.
(2) Agreement
Root port
Designated port

If the upstream device is a third-party device, the rapid state transition implementation might be limited as follows:

- The upstream device uses a rapid transition mechanism similar to that of RSTP.
- The downstream device runs MSTP and does not operate in RSTP mode.

In this case, the following occurs:

1. The root port on the downstream device receives no agreement from the upstream device.
2. It sends no agreement to the upstream device.

As a result, the designated port of the upstream device can transit to the forwarding state only after a period twice the forward delay.

To enable the designated port of the upstream device to transit its state rapidly, enable No Agreement Check on the downstream device's port.

Restrictions and guidelines

Configure No Agreement Check on the root port of your device, because this feature takes effect only if it's configured on root ports.

Prerequisites

Before you configure the No Agreement Check feature, complete the following tasks:

- Connect a device to a third-party upstream device that supports spanning tree protocols through a point-to-point link.
- Configure the same region name, revision level, and VLAN-to-instance mappings on the two devices.

Procedure

Enable the No Agreement Check feature on the root port.

1. Enter system view.
   ```
   system-view
   ```
2. Enter interface view.
   ```
   interface interface-type interface-number
   ```
3. Enable No Agreement Check.
   ```
   stp no-agreement-check
   ```
   By default, No Agreement Check is disabled.

Configuring TC Snooping

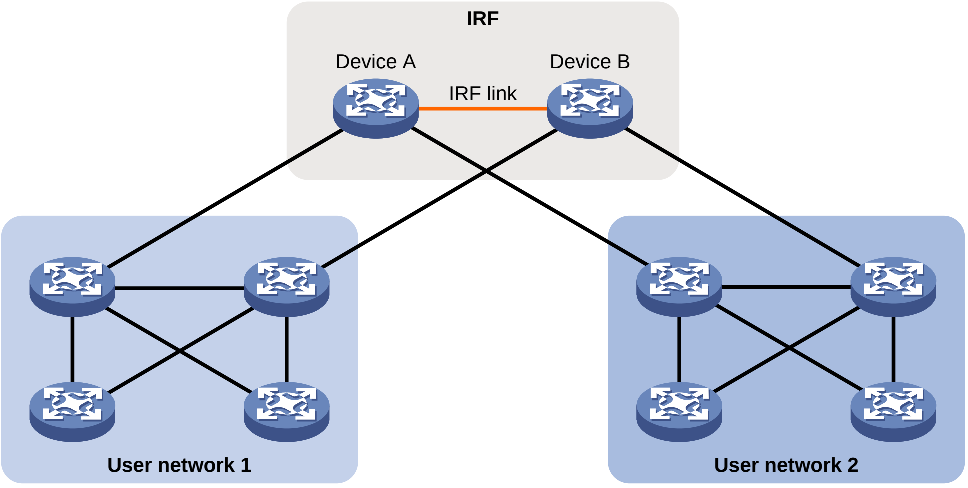
About this task

As shown in Figure 18, Device A and Device B connect to two user networks through double links.

- The spanning tree feature is disabled on Device A and Device B and enabled on all devices in user network 1 and user network 2.
- Device A and Device B transparently transmit BPDUs for both user networks and are not involved in the calculation of spanning trees.

When the network topology changes, it takes time for the IRF fabric to update its MAC address table and ARP table. During this period, traffic in the network might be interrupted.

Figure 18 TC Snooping application scenario



To avoid traffic interruption, you can enable TC Snooping on Device A and Device B. After receiving a TC-BPDU through a port, Device A or Device B updates MAC address table and ARP table entries associated with the port's VLAN. In this way, TC Snooping prevents topology change from interrupting traffic forwarding in the network. For more information about the MAC address table and the ARP table, see "Configuring the MAC address table" and *Layer 3—IP Services Configuration Guide*.

Restrictions and guidelines

- TC Snooping and the spanning tree feature are mutually exclusive. You must globally disable the spanning tree feature before enabling TC Snooping.
- TC Snooping does not support the PVST mode.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Globally disable the spanning tree feature.

   ```
   undo stp global enable
   ```

   By default, the spanning tree feature is disabled globally.

3. Enable TC Snooping.

   ```
   stp tc-snooping
   ```

   By default, TC Snooping is disabled.

Configuring protection features

Spanning tree protection tasks at a glance

All spanning tree protection tasks are optional.

- Configuring BPDU guard
- Enabling root guard
- Enabling loop guard
- Configuring port role restriction
- Configuring TC-BPDU transmission restriction

- Enabling TC-BPDU guard
- Enabling PVST BPDU guard
- About dispute guard

Configuring BPDU guard

About this task

For access layer devices, the access ports can directly connect to the user terminals (such as PCs) or file servers. The access ports are configured as edge ports to allow rapid transition. When these ports receive configuration BPDUs, the system automatically sets the ports as non-edge ports and starts a new spanning tree calculation process. This causes a change of network topology. Under normal conditions, these ports should not receive configuration BPDUs. However, if someone uses configuration BPDUs maliciously to attack the devices, the network will become unstable.

The spanning tree protocol provides the BPDU guard feature to protect the system against such attacks. When edge ports receive configuration BPDUs on a device with BPDU guard enabled, the device performs the following operations:

- Shuts down these ports.
- Notifies the NMS that these ports have been shut down by the spanning tree protocol.

The device reactivates the ports that have been shut down when the port status detection timer expires. You can set this timer by using the **`shutdown-interval`** command. For more information about this command, see device management commands in *Fundamentals Command Reference*.

Restrictions and guidelines

Configure BPDU guard on edge ports which directly connect to a user terminal rather than other device or shared LAN segment.

BPDU guard takes effect only on the edge ports configured by using the **`stp edged-port`** command.

BPDU guard does not take effect on loopback-testing-enabled ports. For more information about loopback testing, see Ethernet interface configuration in *Interface Configuration Guide*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable BPDU guard globally.

   ```
   stp bpdu-protection
   ```

   By default, BPDU guard is globally disabled.

Enabling root guard

About this task

Configure root guard on a designated port.

The root bridge and secondary root bridge of a spanning tree should be located in the same MST region. Especially for the CIST, the root bridge and secondary root bridge are put in a high-bandwidth core region during network design. However, due to possible configuration errors or malicious attacks in the network, the legal root bridge might receive a configuration BPDU with a higher priority. Another device supersedes the current legal root bridge, causing an undesired change of the network topology. The traffic that should go over high-speed links is switched to low-speed links, resulting in network congestion.

To prevent this situation, MSTP provides the root guard feature. If root guard is enabled on a port of a root bridge, this port plays the role of designated port on all MSTIs. After this port receives a configuration BPDU with a higher priority from an MSTI, it performs the following operations:

- Immediately sets that port to the listening state in the MSTI.
- Does not forward the received configuration BPDU.

This is equivalent to disconnecting the link connected to this port in the MSTI. If the port receives no BPDUs with a higher priority within twice the forwarding delay, it reverts to its original state.

Restrictions and guidelines

On a port, the loop guard feature and the root guard feature are mutually exclusive.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable the root guard feature.

   ```
   stp root-protection
   ```

   By default, root guard is disabled.

Enabling loop guard

About this task

Configure loop guard on the root port and alternate ports of a device.

By continuing to receive BPDUs from the upstream device, a device can maintain the state of the root port and blocked ports. However, link congestion or unidirectional link failures might cause these ports to fail to receive BPDUs from the upstream devices. In this situation, the device reselects the following port roles:

- Those ports in forwarding state that failed to receive upstream BPDUs become designated ports.
- The blocked ports transit to the forwarding state.

As a result, loops occur in the switched network. The loop guard feature can suppress the occurrence of such loops.

The initial state of a loop guard-enabled port is **discarding** in every MSTI. When the port receives BPDUs, it transits its state. Otherwise, it stays in the discarding state to prevent temporary loops.

Restrictions and guidelines

Do not enable loop guard on a port that connects user terminals. Otherwise, the port stays in the discarding state in all MSTIs because it cannot receive BPDUs.

On a port, the loop guard feature is mutually exclusive with the root guard feature or the edge port setting.

A loop guard-enabled interface can receive BPDUs and transit from the discarding state to the forwarding state after two forward delays if one of the following events occurs:

- The state of the interface changes from down to up.
- The spanning tree feature is enabled on the up interface.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the loop guard feature.

   stp loop-protection

   By default, loop guard is disabled.

Configuring port role restriction

About this task

Make this configuration on the port that connects to the user access network.

The bridge ID change of a device in the user access network might cause a change to the spanning tree topology in the core network. To avoid this problem, you can enable port role restriction on a port. With this feature enabled, when the port receives a superior BPDU, it becomes an alternate port rather than a root port.

Restrictions and guidelines

Use this feature with caution, because enabling port role restriction on a port might affect the connectivity of the spanning tree topology.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable port role restriction.

   stp role-restriction

   By default, port role restriction is disabled.

Configuring TC-BPDU transmission restriction

About this task

Make this configuration on the port that connects to the user access network.

The topology change to the user access network might cause the forwarding address changes to the core network. When the user access network topology is unstable, the user access network might affect the core network. To avoid this problem, you can enable TC-BPDU transmission restriction on a port. With this feature enabled, when the port receives a TC-BPDU, it does not forward the TC-BPDU to other ports.

Restrictions and guidelines

Enabling TC-BPDU transmission restriction on a port might cause the previous forwarding address table to fail to be updated when the topology changes.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable TC-BPDU transmission restriction.

   stp tc-restriction

By default, TC-BPDU transmission restriction is disabled.

Enabling TC-BPDU guard

About this task

When a device receives topology change (TC) BPDUs (the BPDUs that notify devices of topology changes), it flushes its forwarding address entries. If someone uses TC-BPDUs to attack the device, the device will receive a large number of TC-BPDUs within a short time. Then, the device is busy with forwarding address entry flushing. This affects network stability.

TC-BPDU guard allows you to set the maximum number of immediate forwarding address entry flushes performed within 10 seconds after the device receives the first TC-BPDU. For TC-BPDUs received in excess of the limit, the device performs a forwarding address entry flush when the time period expires. This prevents frequent flushing of forwarding address entries.

Restrictions and guidelines

As a best practice, enable TC-BPDU guard.

Procedure

1. Enter system view.
   `system-view`
2. Enable the TC-BPDU guard feature.
   `stp tc-protection`
   By default, TC-BPDU guard is enabled.
3. (Optional.) Configure the maximum number of forwarding address entry flushes that the device can perform every 10 seconds.
   `stp tc-protection threshold` *`number`*
   The default setting is 6.

Enabling PVST BPDU guard

About this task

This feature takes effect only when the device is operating in MSTP mode.

An MSTP-enabled device forwards PVST BPDUs as data traffic because it cannot recognize PVST BPDUs. If a PVST-enabled device in another independent network receives the PVST BPDUs, a PVST calculation error might occur. To avoid PVST calculation errors, enable PVST BPDU guard on the MSTP-enabled device. The device shuts down a port if the port receives PVST BPDUs.

Procedure

1. Enter system view.
   `system-view`
2. Enable PVST BPDU guard.
   `stp pvst-bpdu-protection`
   By default, PVST BPDU guard is disabled.

About dispute guard

Dispute guard is enabled by default. You do not need to manually configure this feature.

Dispute guard can be triggered by unidirectional link failures. If an upstream port receives inferior BPDUs from a downstream designated port in forwarding or learning state because of a unidirectional link failure, a loop appears. Dispute guard blocks the upstream designated port to prevent the loop.

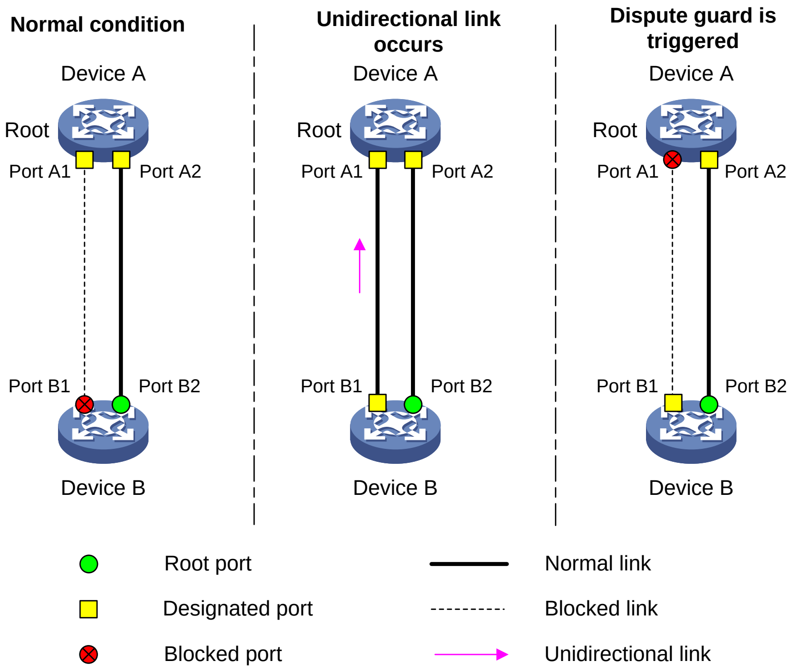
As shown in Figure 19, in normal conditions, the spanning tree calculation result is as follows:

- Device A is the root bridge, and Port A1 is a designated port.
- Port B1 is blocked.

When the link between Port A1 and Port B1 fails in the direction of Port A1 to Port B1 and becomes unidirectional, the following events occur:

1. Port A1 can only receive BPDUs and cannot send BPDUs to Port B1.
2. Port B1 does not receive BPDUs from Port A1 for a certain period of time.
3. Device B determines itself as the root bridge.
4. Port B1 sends its BPDUs to Port A1.
5. Port A1 determines the received BPDUs are inferior to its own BPDUs. A dispute is detected.
6. Dispute guard is triggered and blocks Port A1 to prevent a loop.

Figure 19 Dispute guard triggering scenario



Enabling SNMP notifications for new-root election and topology change events

About this task

This task enables the device to generate logs and report new-root election events or spanning tree topology changes to SNMP. For the event notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see the network management and monitoring configuration guide for the device.

Procedure

1. Enter system view.

```
system-view
```

2. Enable SNMP notifications for new-root election and topology change events.

   snmp-agent trap enable stp [**new-root** | **tc**]

   The default settings are as follows:

   - SNMP notifications are disabled for new-root election events.
   - In MSTP mode, SNMP notifications are enabled in MSTI 0 and disabled in other MSTIs for spanning tree topology changes.
   - In PVST mode, SNMP notifications are disabled for spanning tree topology changes in all VLANs.

Display and maintenance commands for the spanning tree protocols

Execute **display** commands in any view and **reset** command in user view.

Task	Command
Display the spanning tree status and statistics.	**display stp** [**instance** *instance-list* | **vlan** *vlan-id-list*] [**interface** *interface-list* | **slot** *slot-number*] [**brief**]
Display the port role calculation history for the specified MSTI or all MSTIs.	**display stp** [**instance** *instance-list* | **vlan** *vlan-id-list*] **history** [**slot** *slot-number*]
Display the incoming and outgoing TC/TCN BPDU statistics by all ports in the specified MSTI or all MSTIs.	**display stp** [**instance** *instance-list* | **vlan** *vlan-id-list*] **tc** [**slot** *slot-number*]
Display information about ports blocked by spanning tree protection features.	**display stp abnormal-port**
Display BPDU statistics on ports.	**display stp bpdu-statistics** [**interface** *interface-type interface-number* [**instance** *instance-list*]]
Display information about ports shut down by spanning tree protection features.	**display stp down-port**
Display the MST region configuration information that has taken effect.	**display stp region-configuration**
Display the root bridge information of all MSTIs.	**display stp root**
Clear the spanning tree statistics.	**reset stp** [**interface** *interface-list*]

Contents

Configuring LLDP

About LLDP

The Link Layer Discovery Protocol (LLDP) is a standard link layer protocol that allows network devices from different vendors to discover neighbors and exchange system and configuration information.

In an LLDP-enabled network, a device advertises local device information in LLDP Data Units (LLDPDUs) to the directly connected devices. The information distributed through LLDP is stored by its recipients in standard MIBs, making it possible for the information to be accessed by a Network Management System (NMS) through SNMP.

Information that can be distributed through LLDP includes (but is not limited to):

- Major capabilities of the system.
- Management IP address of the system.
- Device ID.
- Port ID.

LLDP agents and bridge modes

An LLDP agent is a mapping of a protocol entity that implements LLDP. Multiple LLDP agents can run on the same interface.

LLDP agents are classified into the following types:

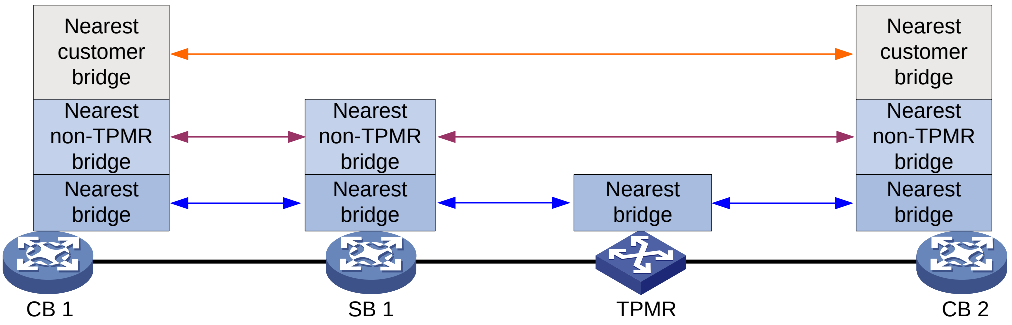
- Nearest bridge agent.
- Nearest customer bridge agent.
- Nearest non-TPMR bridge agent.

  A Two-port MAC Relay (TPMR) is a type of bridge that has only two externally-accessible bridge ports. It supports a subset of the features of a MAC bridge. A TPMR is transparent to all frame-based media-independent protocols except for the following protocols:

  - Protocols destined for the TPMR.
  - Protocols destined for reserved MAC addresses that the relay feature of the TPMR is configured not to forward.

LLDP exchanges packets between neighbor agents and creates and maintains neighbor information for them. Figure 1 shows the neighbor relationships for these LLDP agents.

Figure 1 LLDP neighbor relationships



The types of supported LLDP agents vary with the bridge mode in which LLDP operates. LLDP supports the following bridge modes: customer bridge (CB) and service bridge (SB).

- **Customer bridge mode**—LLDP supports nearest bridge agent, nearest non-TPMR bridge agent, and nearest customer bridge agent. LLDP processes the LLDP frames with destination MAC addresses for these agents and transparently transmits the LLDP frames with other destination MAC addresses in VLANs.
- **Service bridge mode**—LLDP supports nearest bridge agent and nearest non-TPMR bridge agent. LLDP processes the LLDP frames with destination MAC addresses for these agents and transparently transmits the LLDP frames with other destination MAC addresses in VLANs.

LLDP frame formats

LLDP sends device information in LLDP frames. LLDP frames are encapsulated in Ethernet II or Subnetwork Access Protocol (SNAP) format.

LLDP frame encapsulated in Ethernet II

Figure 2 Ethernet II-encapsulated LLDP frame

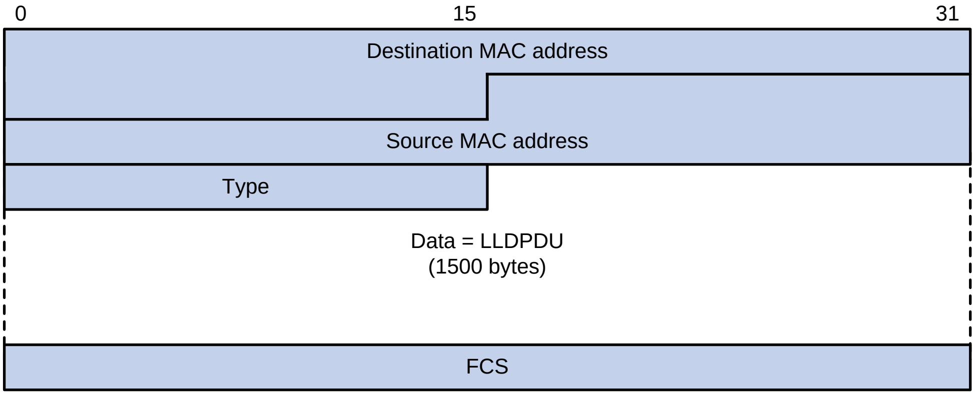


Table 1 Fields in an Ethernet II-encapsulated LLDP frame

Field	Description
Destination MAC address	MAC address to which the LLDP frame is advertised. LLDP specifies different multicast MAC addresses as destination MAC addresses for LLDP frames destined for agents of different types. This helps distinguish between LLDP frames sent and received by different agent types on the same interface. The destination MAC address is fixed to one of the following multicast MAC addresses: • 0x0180-c200-000E for LLDP frames destined for nearest bridge agents. • 0x0180-c200-0000 for LLDP frames destined for nearest customer bridge agents. • 0x0180-c200-0003 for LLDP frames destined for nearest non-TPMR bridge agents.
Source MAC address	MAC address of the sending port.
Type	Ethernet type for the upper-layer protocol. This field is 0x88CC for LLDP.
Data	LLDPDU.
FCS	Frame check sequence, a 32-bit CRC value used to determine the validity of the received Ethernet frame.

LLDP frame encapsulated in SNAP

Figure 3 SNAP-encapsulated LLDP frame

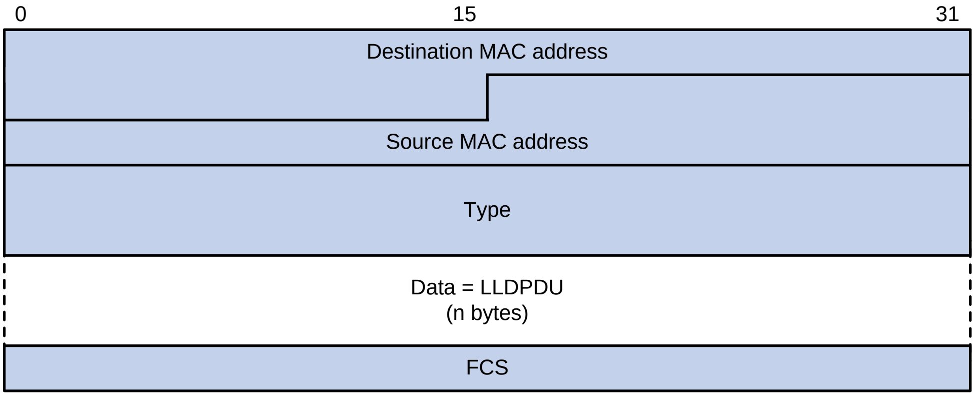


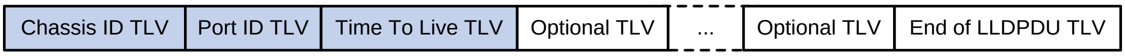
Table 2 Fields in a SNAP-encapsulated LLDP frame

Field	Description
Destination MAC address	MAC address to which the LLDP frame is advertised. It is the same as that for Ethernet II-encapsulated LLDP frames.
Source MAC address	MAC address of the sending port.
Type	SNAP type for the upper-layer protocol. This field is 0xAAAA-0300-0000-88CC for LLDP.
Data	LLDPDU.
FCS	Frame check sequence, a 32-bit CRC value used to determine the validity of the received Ethernet frame.

LLDPDUs

Each LLDP frame contains one LLDPDU. Each LLDPDU is a sequence of type-length-value (TLV) structures.

Figure 4 LLDPDU encapsulation format



As shown in Figure 4, each LLDPDU starts with the following mandatory TLVs: Chassis ID TLV, Port ID TLV, and Time to Live TLV. The mandatory TLVs are followed by a maximum of 29 optional TLVs.

TLVs

A TLV is an information element that contains the type, length, and value fields.

LLDPDU TLVs include the following categories:

- Basic management TLVs.
- Organizationally (IEEE 802.1 and IEEE 802.3) specific TLVs.
- LLDP-MED (media endpoint discovery) TLVs.

Basic management TLVs are essential to device management.

Organizationally specific TLVs and LLDP-MED TLVs are used for enhanced device management. They are defined by standardization or other organizations and are optional for LLDPDUs.

Basic management TLVs

Table 3 lists the basic management TLV types. Some of them are mandatory for LLDPDUs.

Table 3 Basic management TLVs

Type	Description	Remarks
Chassis ID	Specifies the bridge MAC address of the sending device.	Mandatory.
Port ID	Specifies the ID of the sending port: • If the LLDPDU carries LLDP-MED TLVs, the port ID TLV carries the MAC address of the sending port. • Otherwise, the port ID TLV carries the port name.	
Time to Live	Specifies the life of the transmitted information on the receiving device.	
End of LLDPDU	Marks the end of the TLV sequence in the LLDPDU.	
Port Description	Specifies the description for the sending port.	Optional.
System Name	Specifies the assigned name of the sending device.	
System Description	Specifies the description for the sending device.	
System Capabilities	Identifies the primary features of the sending device and the enabled primary features.	
Management Address	Specifies the following elements: • The management address of the local device. • The interface number and object identifier (OID) associated with the address.	

IEEE 802.1 organizationally specific TLVs

Table 4 lists the IEEE 802.1 organizationally specific TLVs.

The device can receive protocol identity TLVs and VID usage digest TLVs, but it cannot send these TLVs.

Layer 3 Ethernet ports support only link aggregation TLVs.

Table 4 IEEE 802.1 organizationally specific TLVs

Type	Description
Port VLAN ID (PVID)	Specifies the port PVID.
Port And Protocol VLAN ID (PPVID)	Indicates whether the device supports protocol VLANs and, if so, what VLAN IDs these protocols will be associated with.
VLAN Name	Specifies the textual name of any VLAN to which the port belongs.
Protocol Identity	Indicates protocols supported on the port.
DCBX	Data center bridging exchange protocol. DCBX TLVs are not supported in the current software version.

Type	Description
EVB module	Edge Virtual Bridging module, including EVB TLV and CDCP TLV. EVB module TLVs are not supported in the current software version.
Link Aggregation	Indicates whether the port supports link aggregation, and if yes, whether link aggregation is enabled.
Management VID	Management VLAN ID.
VID Usage Digest	VLAN ID usage digest.
ETS Configuration	Enhanced Transmission Selection configuration. ETS configuration TLVs are not supported in the current software version.
ETS Recommendation	ETS recommendation. ETS recommendation TLVs are not supported in the current software version.
PFC	Priority-based Flow Control. PFC TLVs are not supported in the current software version.
APP	Application protocol. APP TLVs are not supported in the current software version.
QCN	Quantized Congestion Notification. QCN TLVs are not supported in the current software version.

IEEE 802.3 organizationally specific TLVs

Table 5 shows the IEEE 802.3 organizationally specific TLVs.

The Power Stateful Control TLV is defined in IEEE P802.3at D1.0 and is not supported in later versions. The device sends this type of TLVs only after receiving them.

Table 5 IEEE 802.3 organizationally specific TLVs

Type	Description
MAC/PHY Configuration/Status	Contains the bit-rate and duplex capabilities of the port, support for autonegotiation, enabling status of autonegotiation, and the current rate and duplex mode.
Power Via MDI	Contains the power supply capabilities of the port: • Port class (PSE or PD). • Power supply mode. • Whether PSE power supply is supported. • Whether PSE power supply is enabled. • Whether pair selection can be controlled. • Power supply type. • Power source. • Power priority. • PD requested power. • PSE allocated power.
Maximum Frame Size	Indicates the supported maximum frame size.
Power Stateful Control	Indicates the power state control configured on the sending port, including the following: • Power supply mode of the PSE/PD. • PSE/PD priority. • PSE/PD power.
Energy-Efficient Ethernet	Indicates Energy Efficient Ethernet (EEE). EEE TLVs are not supported in the current software version.

LLDP-MED TLVs

LLDP-MED TLVs provide multiple advanced applications for voice over IP (VoIP), such as basic configuration, network policy configuration, and address and directory management. LLDP-MED TLVs provide a cost-effective and easy-to-use solution for deploying voice devices in Ethernet. LLDP-MED TLVs are shown in Table 6.

If the MAC/PHY configuration/status TLV is not advertisable, none of the LLDP-MED TLVs will be advertised even if they are advertisable.

If the LLDP-MED capabilities TLV is not advertisable, the other LLDP-MED TLVs will not be advertised even if they are advertisable.

Table 6 LLDP-MED TLVs

Type	Description
LLDP-MED Capabilities	Allows a network device to advertise the LLDP-MED TLVs that it supports.
Network Policy	Allows a network device or terminal device to advertise the VLAN ID of a port, the VLAN type, and the Layer 2 and Layer 3 priorities for specific applications.
Extended Power-via-MDI	Allows a network device or terminal device to advertise power supply capability. This TLV is an extension of the Power Via MDI TLV.
Hardware Revision	Allows a terminal device to advertise its hardware version.
Firmware Revision	Allows a terminal device to advertise its firmware version.
Software Revision	Allows a terminal device to advertise its software version.
Serial Number	Allows a terminal device to advertise its serial number.
Manufacturer Name	Allows a terminal device to advertise its vendor name.
Model Name	Allows a terminal device to advertise its model name.
Asset ID	Allows a terminal device to advertise its asset ID. The typical case is that the user specifies the asset ID for the endpoint to facilitate directory management and asset tracking.
Location Identification	Allows a network device to advertise the appropriate location identifier information for a terminal device to use in the context of location-based applications.

Management address

The network management system uses the management address of a device to identify and manage the device for topology maintenance and network management. The management address is encapsulated in the management address TLV.

LLDP operating modes

An LLDP agent can operate in one of the following modes:

- **TxRx mode**—An LLDP agent in this mode can send and receive LLDP frames.
- **Tx mode**—An LLDP agent in this mode can only send LLDP frames.
- **Rx mode**—An LLDP agent in this mode can only receive LLDP frames.

- **Disable mode**—An LLDP agent in this mode cannot send or receive LLDP frames.

Each time the operating mode of an LLDP agent changes, its LLDP protocol state machine reinitializes. A configurable reinitialization delay prevents frequent initializations caused by frequent changes to the operating mode. If you configure the reinitialization delay, an LLDP agent must wait the specified amount of time to initialize LLDP after the LLDP operating mode changes.

Transmitting and receiving LLDP frames

Transmitting LLDP frames

An LLDP agent operating in TxRx mode or Tx mode sends LLDP frames to its directly connected devices both periodically and when the local configuration changes. To prevent LLDP frames from overwhelming the network during times of frequent changes to local device information, LLDP uses the token bucket mechanism to rate limit LLDP frames. For more information about the token bucket mechanism, see *ACL and QoS Configuration Guide*.

LLDP automatically enables the fast LLDP frame transmission mechanism in either of the following cases:

- A new LLDP frame is received and carries device information new to the local device.
- The LLDP operating mode of the LLDP agent changes from Disable or Rx to TxRx or Tx.

The fast LLDP frame transmission mechanism successively sends the specified number of LLDP frames at a configurable fast LLDP frame transmission interval. The mechanism helps LLDP neighbors discover the local device as soon as possible. Then, the normal LLDP frame transmission interval resumes.

Receiving LLDP frames

An LLDP agent operating in TxRx mode or Rx mode confirms the validity of TLVs carried in every received LLDP frame. If the TLVs are valid, the LLDP agent saves the information and starts an aging timer. The initial value of the aging timer is equal to the TTL value in the Time To Live TLV carried in the LLDP frame. When the LLDP agent receives a new LLDP frame, the aging timer restarts. When the aging timer decreases to zero, all saved information ages out.

Protocols and standards

- IEEE 802.1AB-2005, *Station and Media Access Control Connectivity Discovery*
- IEEE 802.1AB-2009, *Station and Media Access Control Connectivity Discovery*
- ANSI/TIA-1057, *Link Layer Discovery Protocol for Media Endpoint Devices*
- IEEE Std 802.1Qaz-2011, *Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks-Amendment 18: Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes*

Restrictions and guidelines: LLDP configuration

When you configure LLDP, follow these restrictions and guidelines:

- Some of the LLDP configuration tasks are available in different interface views (see Table 7).

Table 7 Support of LLDP configuration tasks in different views

Tasks	Supported views
Enabling LLDP	Layer 2 Ethernet interface view Layer 3 Ethernet interface view
Setting the LLDP operating mode	

Tasks	Supported views
Configuring the advertisable TLVs	Layer 2 aggregate interface view Layer 3 aggregate interface view
Configuring advertisement of the management address TLV	
Setting the encapsulation format for LLDP frames	
Enabling LLDP polling	
Configuring LLDP trapping and LLDP-MED trapping	

LLDP tasks at a glance

To configure LLDP, perform the following tasks:

1. Enabling LLDP
2. Setting the LLDP bridge mode
3. Setting the LLDP operating mode
4. (Optional.) Setting the LLDP reinitialization delay
5. (Optional.) Configuring LLDP packet-related settings
    - Configuring the advertisable TLVs
    - Configuring advertisement of the management address TLV
    - Setting the encapsulation format for LLDP frames
    - Setting LLDP frame transmission parameters
    - Setting the timeout for receiving LLDP frames
6. (Optional.) Enabling LLDP polling
7. (Optional.) Disabling LLDP PVID inconsistency check
8. (Optional.) Configuring LLDP trapping and LLDP-MED trapping
9. (Optional.) Setting the source MAC address of LLDP frames
10. (Optional.) Enabling generation of ARP or ND entries for received management address TLVs

Enabling LLDP

Restrictions and guidelines

For LLDP to take effect on specific ports, you must enable LLDP both globally and on these ports.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enable LLDP globally.
   ```
   lldp global enable
   ```
   By default, LLDP is disabled globally.
3. Enter interface view.
   ```
   interface interface-type interface-number
   ```
4. Enable LLDP.
   ```
   lldp enable
   ```
   By default, LLDP is enabled on a port.

Setting the LLDP bridge mode

1. Enter system view.

   system-view

2. Set the LLDP bridge mode.

   - Set the LLDP bridge mode to service bridge.

     lldp mode service-bridge

     By default, LLDP operates in customer bridge mode.

   - Set the LLDP bridge mode to customer bridge.

     undo lldp mode service-bridge

   By default, LLDP operates in customer bridge mode.

Setting the LLDP operating mode

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the LLDP operating mode.

   - In Layer 2 or Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **admin-status** { **disable** | **rx** | **tx** | **txrx** }

     In Ethernet interface view, if you do not specify an agent type, the command sets the operating mode for the nearest bridge agent.

   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **admin-status** { **disable**| **rx**| **tx**| **txrx** }

     In aggregate interface view, you can set the operating mode only for the nearest customer bridge agent and nearest non-TPMR bridge agent.

   By default:

   - The nearest bridge agent operates in TxRx mode.
   - The nearest customer bridge agent and nearest non-TPMR bridge agent operate in Disable mode.

Setting the LLDP reinitialization delay

About this task

When the LLDP operating mode changes on a port, the port initializes the protocol state machines after an LLDP reinitialization delay. By adjusting the delay, you can avoid frequent initializations caused by frequent changes to the LLDP operating mode on a port.

Procedure

1. Enter system view.

   system-view

2. Set the LLDP reinitialization delay.

```
lldp timer reinit-delay delay
```

The default LLDP reinitialization delay is 2 seconds.

Configuring the advertisable TLVs

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Configure the advertisable TLVs.

   - In Layer 2 Ethernet interface view:

```
lldp tlv-enable { basic-tlv { all | port-description |
system-capability | system-description | system-name |
management-address-tlv [ ipv6 ] [ ip-address ] } | dot1-tlv { all |
port-vlan-id | link-aggregation | protocol-vlan-id [ vlan-id ] |
vlan-name [ vlan-id ] | management-vid [ mvlan-id ] } | dot3-tlv { all |
mac-physic | max-frame-size | power } | med-tlv { all | capability |
inventory | network-policy [ vlan-id ] | power-over-ethernet |
location-id { civic-address device-type country-code { ca-type
ca-value }&<1-10> | elin-address tel-number } } }
```

     By default, the nearest bridge agent advertises all supported TLVs except the following TLVs:

     – Location identification TLVs.

     – Port and protocol VLAN ID TLVs.

     – VLAN name TLVs.

     – Management VLAN ID TLVs.

```
lldp agent nearest-nontpmr tlv-enable { basic-tlv { all |
port-description | system-capability | system-description |
system-name | management-address-tlv [ ipv6 ] [ ip-address ] } |
dot1-tlv { all | port-vlan-id | link-aggregation } }
```

```
lldp tlv-enable dot1-tlv { protocol-vlan-id [ vlan-id ] | vlan-name
[ vlan-id ] | management-vid [ mvlan-id ] }
```

     By default, the nearest non-TPMR bridge agent does not advertise any TLVs.

```
lldp agent nearest-customer tlv-enable { basic-tlv { all |
port-description | system-capability | system-description |
system-name | management-address-tlv [ ipv6 ] [ ip-address ] } |
dot1-tlv { all | port-vlan-id | link-aggregation } }
```

```
lldp tlv-enable dot1-tlv { protocol-vlan-id [ vlan-id ] | vlan-name
[ vlan-id ] | management-vid [ mvlan-id ] }
```

     By default, the nearest customer bridge agent advertises all the supported basic management TLVs and IEEE 802.1 organizationally specific TLVs.

   - In Layer 3 Ethernet interface view:

```
lldp tlv-enable { basic-tlv { all | port-description |
system-capability | system-description | system-name |
management-address-tlv [ ipv6 ] [ ip-address | interface loopback
interface-number ] } | dot1-tlv { all | link-aggregation } | dot3-tlv
{ all | mac-physic | max-frame-size | power } | med-tlv { all |
capability | inventory | power-over-ethernet | location-id
```

{ **civic-address** *device-type country-code* { *ca-type ca-value* }&<1-10> | **elin-address** *tel-number* } } }

By default, the nearest bridge agent advertises the following TLVs:

- Link aggregation TLVs in the 802.1 organizationally specific TLV set.
- All supported 802.3 organizationally specific TLVs.
- All supported LLDP-MED TLVs except the network policy TLVs.

lldp agent { **nearest-nontpmr** | **nearest-customer** } **tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **link-aggregation** } }

By default:

- The nearest non-TPMR bridge agent does not advertise any TLVs.
- The nearest customer bridge agent advertises all supported basic management TLVs and link aggregation TLVs in the IEEE 802.1 organizationally specific TLV set.

- In Layer 2 aggregate interface view:

  lldp tlv-enable dot1-tlv { **protocol-vlan-id** [*vlan-id*] | **vlan-name** [*vlan-id*] | **management-vid** [*mvlan-id*]

  lldp agent nearest-nontpmr tlv-enable { **basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** } | **dot1-tlv** { **all** | **port-vlan-id** } }

  By default, the nearest non-TPMR bridge agent does not advertise any TLVs.

  lldp agent nearest-customer tlv-enable { **basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** } | **dot1-tlv** { **all** | **port-vlan-id** } }

  By default, the nearest customer bridge agent advertises all supported basic management TLVs and the following IEEE 802.1 organizationally specific TLVs:

  - Port and protocol VLAN ID TLVs.
  - VLAN name TLVs.
  - Management VLAN ID TLVs.

  The nearest bridge agent is not supported.

- In Layer 3 aggregate interface view:

  lldp agent { **nearest-customer** | **nearest-nontpmr** } **tlv-enable basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** }

  By default:

  - The nearest non-TPMR bridge agent does not advertise any TLVs.
  - The nearest customer bridge agent advertises all supported basic management TLVs.

  The nearest bridge agent is not supported.

Configuring advertisement of the management address TLV

About this task

LLDP encodes management addresses in numeric or string format in management address TLVs.

If a neighbor encodes its management address in string format, set the encoding format of the management address to **string** on the connecting port. This guarantees normal communication with the neighbor.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable advertisement of the management address TLV on the interface and set the management address to be advertised.

   - In Layer 2 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*]

   - In Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*] | **interface loopback** *interface-number*]

   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*]

4. Set the encoding format of the management address to string.

   - In Layer 2 or Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **management-address-format string**

   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **management-address-format string**

   The default management address encoding format is numeric.

Setting the encapsulation format for LLDP frames

About this task

Earlier versions of LLDP require the same encapsulation format on both ends to process LLDP frames. To successfully communicate with a neighboring device running an earlier version of LLDP, the local device must be set with the same encapsulation format.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Set the encapsulation format for LLDP frames to SNAP.
   - In Layer 2 or Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **encapsulation snap**
   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **encapsulation snap**

   By default, the Ethernet II encapsulation format is used.

Setting LLDP frame transmission parameters

About this task

The Time to Live TLV carried in an LLDPDU determines how long the device information carried in the LLDPDU can be saved on a recipient device.

By setting the TTL multiplier, you can configure the TTL of locally sent LLDPDUs. The TTL is expressed by using the following formula:

TTL = Min (65535, (TTL multiplier × LLDP frame transmission interval + 1))

As the expression shows, the TTL can be up to 65535 seconds. TTLs greater than 65535 will be rounded down to 65535 seconds.

Procedure

1. Enter system view.

   system-view
2. Set the TTL multiplier.

   lldp hold-multiplier *value*

   The default setting is 4.
3. Set the LLDP frame transmission interval.

   lldp timer tx-interval *interval*

   The default setting is 30 seconds.
4. Set the token bucket size for sending LLDP frames.

   lldp max-credit *credit-value*

   The default setting is 5.
5. Set the number of LLDP frames sent each time fast LLDP frame transmission is triggered.

   lldp fast-count *count*

   The default setting is 4.
6. Set the fast LLDP frame transmission interval.

   lldp timer fast-interval *interval*

   The default setting is 1 second.

Setting the timeout for receiving LLDP frames

About this task

This feature allows the device to detect the presence of directly connected neighbors by setting the timeout timer for receiving LLDP frames. If an interface has not received any frames when the timeout timer expires, the device reports a no LLDP neighbor event to the NETCONF module.

Restrictions and guidelines

To avoid misdetection, make sure the timeout for receiving LLDP frames is greater than the LLDP frame transmission interval.

Procedure

1. Enter system view.

   system-view

2. Set the timeout for receiving LLDP frames.

   lldp timer rx-timeout *timeout*

   By default, no timeout is set for receiving LLDP frames, and the device does not report no LLDP neighbor events.

Enabling LLDP polling

About this task

With LLDP polling enabled, a device periodically searches for local configuration changes. When the device detects a configuration change, it sends LLDP frames to inform neighboring devices of the change.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable LLDP polling and set the polling interval.
   - In Layer 2 or Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **check-change-interval** *interval*

   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **check-change-interval** *interval*

   By default, LLDP polling is disabled.

Disabling LLDP PVID inconsistency check

About this task

By default, when the system receives an LLDP packet, it compares the PVID value contained in the packet with the PVID configured on the receiving interface. If the two PVIDs do not match, a log message will be printed to notify the user.

You can disable PVID inconsistency check if different PVIDs are required on a link.

Procedure

1. Enter system view.

   `system-view`

2. Disable LLDP PVID inconsistency check.

   `lldp ignore-pvid-inconsistency`

   By default, LLDP PVID inconsistency check is enabled.

Configuring CDP compatibility

About this task

To enable your device to exchange information with a directly connected Cisco device that supports only CDP, you must enable CDP compatibility.

CDP compatibility enables your device to receive and recognize CDP packets from the neighboring CDP device and send CDP packets to the neighboring device. The CDP packets sent to the neighboring CDP device carry the following information:

- Device ID.
- ID of the port connecting to the neighboring device.
- Port IP address.
- TTL.

The port IP address is the primary IP address of a VLAN interface in up state. The VLAN ID of the VLAN interface must be the lowest among the VLANs permitted on the port. If no VLAN interfaces of the permitted VLANs are assigned an IP address or all VLAN interfaces are down, no port IP address will be advertised.

You can view the neighboring CDP device information that can be recognized by the device in the output of the **`display lldp neighbor-information`** command. For more information about the **`display lldp neighbor-information`** command, see LLDP commands in *Layer 2—LAN Switching Command Reference*.

CDP-compatible LLDP operates in one of the following modes:

- **TxRx**—CDP packets can be transmitted and received.
- **Disable**—CDP packets cannot be transmitted or received.

Restrictions and guidelines

When you configure CDP compatibility for LLDP, follow these restrictions and guidelines:

- To make CDP-compatible LLDP take effect on a port, follow these steps:
  - **a.** Enable CDP-compatible LLDP globally.
  - **b.** Configure CDP-compatible LLDP to operate in TxRx mode on the port.
- The maximum TTL value that CDP allows is 255 seconds. To make CDP-compatible LLDP work correctly with Cisco IP phones, configure the LLDP frame transmission interval to be no more than 1/3 of the TTL value.

Prerequisites

Before you configure CDP compatibility, complete the following tasks:

- Globally enable LLDP.
- Enable LLDP on the port connecting to a CDP device.
- Configure LLDP to operate in TxRx mode on the port.

Procedure

1. Enter system view.

   system-view

2. Enable CDP compatibility globally.

   lldp compliance cdp

   By default, CDP compatibility is disabled globally.

3. Enter Layer 2/Layer 3 Ethernet interface view or management Ethernet interface view.

   interface *interface-type interface-number*

4. Configure CDP-compatible LLDP to operate in TxRx mode.

   lldp compliance admin-status cdp txrx

   By default, CDP-compatible LLDP operates in **disable** mode.

Configuring LLDP trapping and LLDP-MED trapping

About this task

LLDP trapping or LLDP-MED trapping notifies the network management system of events such as newly detected neighboring devices and link failures.

To prevent excessive LLDP traps from being sent when the topology is unstable, set a trap transmission interval for LLDP.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable LLDP trapping.

   - In Layer 2 or Layer 3 Ethernet interface view:

     lldp [**agent** { **nearest-customer** | **nearest-nontpmr** }] **notification remote-change enable**

   - In Layer 2/Layer 3 aggregate interface view:

     lldp agent { **nearest-customer** | **nearest-nontpmr** } **notification remote-change enable**

   By default, LLDP trapping is disabled.

4. (In Layer 2 or Layer 3 Ethernet interface view.) Enable LLDP-MED trapping.

   lldp notification med-topology-change enable

   By default, LLDP-MED trapping is disabled.

5. Return to system view.

   quit

6. (Optional.) Set the LLDP trap transmission interval.

   lldp timer notification-interval *interval*

   The default setting is 30 seconds.

Setting the source MAC address of LLDP frames

About this task

Use this feature together with generation of ARP or ND entries for received management address TLVs for the device to use the MAC address of a Layer 3 Ethernet subinterface as the source MAC address of LLDP frames. This ensures that the LLDP neighbor can learn correct ARP or ND entries.

This feature allows you to set the source MAC address of LLDP frames to the MAC address of a Layer 3 Ethernet subinterface that terminates a specified VLAN. If the VLAN is not terminated by any Layer 3 Ethernet subinterface, the MAC address of the Layer 3 Ethernet interface is used as the source MAC address of outgoing LLDP frames. For more information about VLAN termination, see "Configuring VLAN termination."

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Set the source MAC address of LLDP frames to the MAC address of the Layer 3 Ethernet subinterface that terminates a specified VLAN.

   `lldp source-mac vlan` *`vlan-id`*

   By default, the source MAC address of LLDP frames is the MAC address of the egress interface.

Enabling generation of ARP or ND entries for received management address TLVs

About this task

This feature enables the device to generate an ARP or ND entry after receiving an LLDP frame containing a management address TLV on an interface. The ARP or ND entry maps the advertised management address to the source MAC address of the frame.

You can enable generation of both ARP and ND entries on an interface. If the management address TLV contains an IPv4 address, the device generates an ARP entry. If the management address TLV contains an IPv6 address, the device generates an ND entry.

Use this feature together with the source MAC address configuration for LLDP frames for the device to use the MAC address of a Layer 3 Ethernet subinterface as the source MAC address of LLDP frames. This ensures that the LLDP neighbor can learn correct ARP or ND entries.

If you specify a VLAN ID, the Layer 3 Ethernet subinterface that terminates the VLAN is recorded as the output interface in the generated ARP or ND entries. If the VLAN is not terminated by any Layer 3 Ethernet subinterface or no VLAN is specified, the Layer 3 Ethernet interface is recorded as the output interface. For more information about VLAN termination, see "Configuring VLAN termination."

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Enable generation of ARP or ND entries for management address TLVs received on the interface.

lldp management-address { **arp-learning** | **nd-learning** } [**vlan** *vlan-id*]

By default, generation of ARP or ND entries for received management address TLVs is disabled on an interface.

You can enable generation of both ARP and ND entries on an interface.

Display and maintenance commands for LLDP

Execute **display** commands in any view.

Task	Command
Display local LLDP information.	**display lldp local-information** [**global** | **interface** *interface-type interface-number*]
Display the information contained in the LLDP TLVs sent from neighboring devices.	**display lldp neighbor-information** [[[**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }] [**verbose**]] | **list** [**system-name** *system-name*]]
Display LLDP statistics.	**display lldp statistics** [**global** | [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]]
Display LLDP status of a port.	**display lldp status** [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]
Display types of advertisable optional LLDP TLVs.	**display lldp tlv-config** [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]

Contents

Configuring normal Layer 2 forwarding

About normal Layer 2 forwarding

When an incoming frame's destination MAC address does not match any Layer 3 interface's MAC address, normal Layer 2 forwarding forwards the frame through a Layer 2 interface.

The device uses the destination MAC address of the frame to look for a match in the MAC address table.

- The device forwards the frame out of the outgoing interface in the matching entry if a match is found.
- The device floods the frame to all interfaces in the VLAN of the frame if no match is found.

Normal Layer 2 forwarding is enabled by default.

Display and maintenance commands for Layer 2 forwarding

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display Layer 2 forwarding statistics.	**display mac-forwarding statistics** [**interface** *interface-type interface-number*]
Clear Layer 2 forwarding statistics.	**reset mac-forwarding statistics**

Configuring fast Layer 2 forwarding

About fast Layer 2 forwarding

Fast Layer 2 forwarding improves packet forwarding efficiency by using a high-speed cache and flow-based technology. It identifies a flow by using the following items:

- Source IP address.
- Source port number.
- Destination IP address.
- Destination port number.
- Protocol number.
- Input interface.
- Output interface.
- VLAN ID.

Fast Layer 2 forwarding creates an entry in a high-speed cache by obtaining the forwarding information of a flow's first packet. Subsequent packets of the flow are forwarded based on the entry.

Fast Layer 2 forwarding is enabled by default.

Disabling VLAN ID check for fast Layer 2 forwarding

About this task

The VLAN ID of a packet helps the device to determine the TCP session to which the packet belongs. On a hot backup system formed by two firewalls, you must disable VLAN ID check if the traffic incoming interfaces on the primary and secondary devices belong to different VLANs. If you enable VLAN ID check, traffic cannot match session entries correctly when asymmetric-path traffic exists.

Procedure

1. Enter system view.
   system-view
2. Disable VLAN ID check for fast Layer 2 forwarding.
   undo mac fast-forwarding check-vlan-id
   By default, VLAN ID check is enabled for fast Layer 2 forwarding.

Display and maintenance commands for fast Layer 2 forwarding

Execute **display** commands in any view.

Task	Command
Display IPv4 fast forwarding entries.	**display mac-forwarding cache ip** [*ip-address*] [**slot** *slot-number*]

Display IPv4 fast forwarding entries for fragments.	**display mac-forwarding cache ip fragment** [*ip-address*] [**slot** *slot-number*]
Display IPv6 fast forwarding entries.	**display mac-forwarding cache ipv6** [*ipv6-address*] [**slot** *slot-number*]

Configuring bridge forwarding

About bridge forwarding

Bridge forwarding allows users to customize bridge instances to implement VLAN or port based secure packet forwarding.

Bridge forwarding types

Bridge forwarding has the following types:

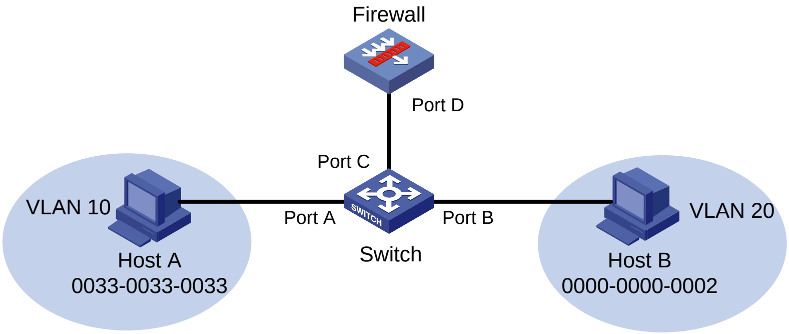
- **Inter-VLAN bridge forwarding**—Forwards a packet between different VLANs.
- **Inline forwarding**—Inline forwarding has the following types:
  - **Reflect-type bridge forwarding**—Forwards a packet through the receiving port of the packet.
  - **Forward-type bridge forwarding**—Forwards a packet through a port that is different from the receiving port of the packet.
  - **Blackhole-type bridge forwarding**—Drops the received packets.

Inter-VLAN bridge forwarding

Inter-VLAN bridge forwarding enables communication between different VLANs at the data link layer. It is typically used on firewall products. A firewall connected to a switch filters Layer 2 traffic before passing the traffic to the switch for further forwarding.

As shown in Figure 1, bridge forwarding enables communication between VLANs 10 and 20. VLANs 10 and 20 are in bridge instance 1 on the firewall. The interface that connects the switch to the firewall is Port C.

Figure 1 Bridge forwarding



The following process uses ARP to describe the MAC address learning and packet forwarding in bridge forwarding. Host A requires the MAC address of Host B and sends out an ARP request. When receiving the request from Host A, bridge forwarding processes the request as follows:

1. The switch performs the following operations:
   a. Learns a new entry to the MAC address table of the switch. The entry contains the MAC address of Host A (0033-0033-0033), the output interface Port A, and VLAN 10.
   b. Broadcasts the request in VLAN 10. Because VLAN 10 is in bridge instance 1, the request enters the firewall through Port C.
2. The firewall performs the following operations:

   a. Learns a new entry to the MAC address table of bridge instance 1. The entry contains the MAC address of Host A (0033-0033-0033), the output interface Port D, and VLAN 10.
   b. Replaces the VLAN tag of the request with VLAN 20 and broadcasts the request in VLAN 20. No matching MAC address entry exists in VLAN 20.
   c. Sends the request to the switch through Port D.
3. The switch performs the following operations:
   a. Learns a new entry to the MAC address table of the switch. The entry contains the MAC address of Host A (0033-0033-0033), interface Port C, and VLAN 20.
   b. Broadcasts the request in VLAN 20.

Host B in VLAN 20 receives the request, places its MAC address in the reply, and sends the reply to Host A. Bridge forwarding processes the reply as follows:

1. The switch performs the following operations:
   a. Learns a new entry to the MAC address table of the switch. The entry contains the MAC address of Host B (0000-0000-0002), the output interface Port B, and VLAN 20.
   b. Uses the destination MAC address 0033-0033-0033 and VLAN ID 20 to search the MAC address table for a match. An entry with interface Port C is found.
   c. Sends the reply to the firewall through Port C.
2. The firewall performs the following operations:
   a. Learns a new entry to the MAC address table of bridge instance 1. The entry contains the MAC address of Host B (0000-0000-0002), the output interface Port D, and VLAN 20.
   b. Uses the destination MAC address 0033-0033-0033 to search the MAC address table of bridge instance 1 for a match. An entry with the output interface Port D and VLAN 10 is found.
   c. Replaces the VLAN ID of the reply (VLAN 20) with the VLAN ID in the entry (VLAN 10).
   d. Sends the reply to the switch through Port D.
3. The switch performs the following operations:
   a. Uses the destination MAC address 0033-0033-0033 and VLAN ID 10 to search the MAC address table for a match. An entry with the output interface Port A exists.
   b. Forwards the reply through Port A.

Inline forwarding

Inline forwarding monitors traffic at the data link layer. It is typically used on security devices. Layer 2 traffic arriving at a device is redirected to a security device based on QoS policies, filtered, and then forwarded toward the destination.

Inline forwarding can be further classified into the following forwarding types:

- Reflect-type bridge forwarding.
- Blackhole-type bridge forwarding.
- Forward-type bridge forwarding.

Reflect-type/blackhole-type bridge forwarding

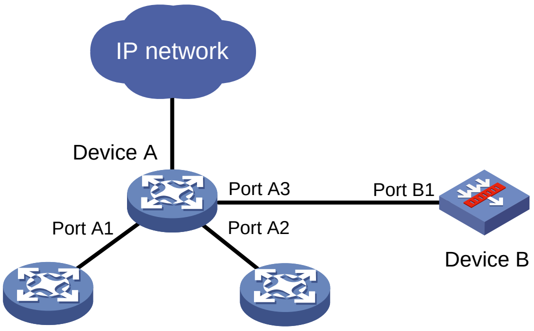
Reflect-type bridge forwarding and blackhole-type bridge forwarding are applicable to the scenario where a device directly accesses the network and is directly connected to a security device.

As shown in Figure 2, Device A is connected to the security device (Device B) through a physical port.

- In reflect-type bridge forwarding mode, packets arriving at Device A are forwarded to Device B for security service processing and then sent back to Device A for forwarding.

- In blackhole-type bridge forwarding mode, packets arriving at Device A are forwarded to Device B. Device B processes the packets and then drops the packets.

Figure 2 Reflect-type/blackhole-type bridge forwarding network

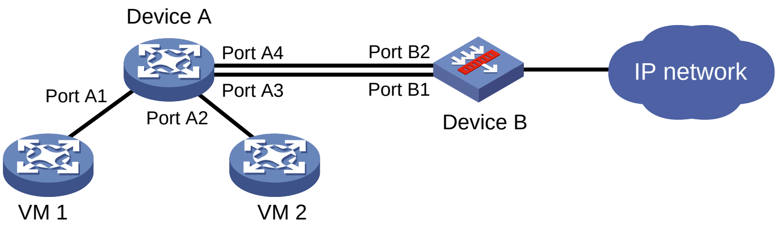


Forward-type bridge forwarding

Forward-type bridge forwarding is applicable to the scenario where a device accesses the network through a security device.

As shown in Figure 3, Device A is connected to Device B through two physical ports. Device B uses one port to receive packets from Device A, and it uses the other port to send packets back to Device A.

Figure 3 Forward-type bridge forwarding network



Packet processing example in inline forwarding

As shown in Figure 2 and Figure 3, when VMs 1 and 2 communicate through Device A, inline forwarding processes packets between them as follows:

- Device A forwards the received packets to Device B.
- Device B passes the IP packets to the security modules for processing and sends other types of packets back to Device A.
- Device B creates forwarding entries for IP packets that meet the security requirements and forwards them to Device A. IP packets that do not meet the security requirements are dropped.

Configuring bridging forwarding

Configuring inter-VLAN bridge forwarding

1. Enter system view.

   system-view

2. (Optional.) Set the aging timer for dynamic MAC address entries.

   bridge mac-address timer aging *seconds*

   The default setting is 300 seconds.

3. Create an inter-VLAN bridge instance, and enter bridge view.

bridge *bridge-index* **inter-vlan**

4. Add a list of VLANs to the bridge instance.

   add vlan *vlan-id-list*

5. (Optional.) Set the MAC learning limit on the bridge instance.

   mac-address max-mac-count *count*

   By default, a maximum of 4096 MAC addresses can be learned on a bridge instance.

Configuring inline forwarding

Restrictions and guidelines

You can manually create reflect-type, forward-type, and blackhole-type bridge instances for inline forwarding and add interfaces to the instances.

The device will automatically create a forward-type bridge instance upon insertion of a hardware bypass subcard. For a forward-type bridge instance to be automatically created, make sure the device does not have an inter-VLAN bridge instance before you insert a hardware bypass subcard.

If you configure inline forwarding on a security device connected to a switch, disable MAC address learning on the switch's interface that is connected to the security device to avoid frequent MAC moves.

Only one interface can be added to a reflect-type or blackhole-type bridge instance.

Only two interfaces can be added to a manually created forward-type bridge instance. The two interfaces must be the same type.

An automatically created forward-type bridge instance uses the pair of interfaces on the bypass subcard by default and you cannot edit the interfaces in the instance.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure the device to ignore the tunnel encapsulation when forwarding tunneled packets in inline mode.

   bridge tunnel-encapsulation skip

   In inline forwarding mode, tunneled packets are forwarded based on information in the tunnel encapsulation by default.

3. Create a bridge instance and enter its view.

   - Create a reflect-type bridge instance.

     bridge *bridge-index* **reflect**

   - Create a forward-type bridge instance.

     bridge *bridge-index* **forward**

   - Create a blackhole-type bridge instance.

     bridge *bridge-index* **blackhole**

4. Add an interface to the bridge instance.

   add interface *interface-type interface-number*

   By default, no interfaces exist in a manually created bridge instance.

Configuring security service bypass

About this task

By default, packets are processed by the security service first before being forwarded according to the configured bridge forwarding mode.

The security service bypass feature enables user traffic to bypass security service processing of a security device and be forwarded directly according to the configured bridge forwarding mode.

The device supports only the internal bypass mode. In internal bypass mode, user traffic is sent to the security device but is not processed by it. The security device directly forwards or drops the traffic according to the configured bridge forwarding mode.

Restrictions and guidelines for security service bypass

If you configure the **bypass enable** command for the same bridge instance multiple times, the most recent configuration takes effect.

Enabling internal security service bypass

1. Enter system view.
   system-view
2. Enter bridge instance view.
   - Enter the view of a reflect-type bridge instance.
     bridge *bridge-index* **reflect**
   - Enter the view of an automatically created forward-type bridge instance.
     bridge *bridge-index* **forward**
   - Enter the view of a manually created forward-type bridge instance.
     bridge *bridge-index* **forward**
   - Enter the view of a blackhole-type bridge instance.
     bridge *bridge-index* **blackhole**
3. Enable internal security service bypass.
   bypass enable
   Security service bypass is disabled by default.

Display and maintenance commands for bridge forwarding

Execute **display** commands in any view.

Task	Command
Display MAC address entries in bridge instances.	**display bridge mac-address** [*bridge-index* [**vlan** *vlan-id*]] [**count**] [**slot** *slot-number*]

Configuring fast bridge forwarding

About fast bridge forwarding

Fast bridge forwarding improves packet forwarding efficiency by using a high-speed cache and flow-based technology. It identifies a flow by using the following items:

- Source IP address.
- Source port number.
- Destination IP address.
- Destination port number.
- Protocol number.
- Input interface.
- Output interface.
- VLAN ID.

Fast bridge forwarding creates an entry in a high-speed cache by obtaining the forwarding information of a flow's first packet. Subsequent packets of the flow are forwarded based on the entry.

Fast bridge forwarding is enabled by default.

Disabling VLAN ID check for fast bridge forwarding

About this task

The VLAN ID of a packet helps the device to determine the TCP session to which the packet belongs. On a hot backup system formed by two firewalls, you must disable VLAN ID check if the traffic incoming interfaces on the primary and secondary devices belong to different VLANs. If you enable VLAN ID check, traffic cannot match session entries correctly when asymmetric-path traffic exists.

Restrictions and guidelines

Only inline forwarding supports VLAN ID check. You do not need to configure this feature for inter-VLAN fast bridge forwarding because the device does not check VLAN IDs for inter-VLAN fast bridge forwarding. That is, this feature does not take effect on inter-VLAN fast bridge forwarding.

On a hot backup system formed by two firewalls, inter-VLAN fast bridge forwarding enables a packet to match the same session after being transmitted between the primary and secondary devices.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Disable VLAN ID check for fast bridge forwarding.
   ```
   undo bridge fast-forwarding check-vlan-id
   ```
   By default, VLAN ID check is enabled for fast bridge forwarding.

Display and maintenance commands for fast bridge forwarding

Execute **display** commands in any view.

Task	Command
Display IPv4 fast bridge forwarding entries.	**display bridge cache ip** { **inline** | **inter-vlan** } [*ip-address*] [**slot** *slot-number*]
Display IPv4 fast bridge forwarding entries for fragments.	**display bridge cache ip fragment** { **inline** | **inter-vlan** } [*ip-address*] [**slot** *slot-number*]
Display IPv6 fast bridge forwarding entries.	**display bridge cache ipv6** { **inline** | **inter-vlan** } [*ipv6-address*] [**slot** *slot-number*]

NSFOCUS Firewall Series

NF Layer 2—WAN Access

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for Layer 2 WAN access features, including PPP and Mobile communication modem.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

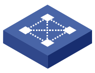
GUI conventions

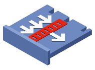
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring PPP

About PPP

Point-to-Point Protocol (PPP) is a point-to-point link layer protocol. It provides user authentication, supports synchronous/asynchronous communication, and allows for easy extension.

PPP protocols

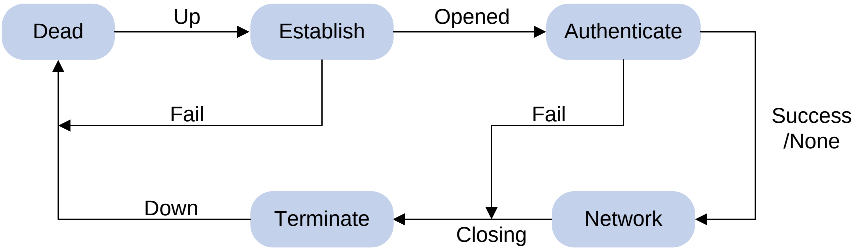
PPP includes the following protocols:

- **Link control protocol (LCP)**—Establishes, tears down, and monitors data links.
- **Network control protocol (NCP)**—Negotiates the packet format and type for data links.
- **Authentication protocols**—Authenticate users. Protocols include the following:
  - Password Authentication Protocol (PAP).
  - Challenge Handshake Authentication Protocol (CHAP).
  - Microsoft CHAP (MS-CHAP).
  - Microsoft CHAP Version 2 (MS-CHAP-V2).

PPP link establishment process

Figure 1 shows the PPP link establishment process.

Figure 1 PPP link establishment process



1. Initially, PPP is in Link Dead phase. After the physical layer goes up, PPP enters the Link Establishment phase (Establish).
2. In the Link Establishment phase, the LCP negotiation is performed. The LCP configuration options include Authentication-Protocol, Maximum-Receive-Unit (MRU), Magic-Number, Protocol-Field-Compression (PFC), Address-and-Control-Field-Compression (ACFC).
   - If the negotiation fails, LCP reports a Fail event, and PPP returns to the Dead phase.
   - If the negotiation succeeds, LCP enters the Opened state and reports an Up event, indicating that the underlying layer link has been established. At this time, the PPP link is not established for the network layer, and network layer packets cannot be transmitted over the link.
3. If authentication is configured, the PPP link enters the Authentication phase, where PAP, CHAP, MS-CHAP, or MS-CHAP-V2 authentication is performed.
   - If the client fails to pass the authentication, LCP reports a Fail event and enters the Link Termination phase. In this phase, the link is torn down and LCP goes down.
   - If the client passes the authentication, LCP reports a Success event.

4. If a network layer protocol is configured, the PPP link enters the Network-Layer Protocol phase for NCP negotiation, such as IPCP negotiation and IPv6CP negotiation.
   - If the NCP negotiation succeeds, the link goes up and becomes ready to carry negotiated network-layer protocol packets.
   - If the NCP negotiation fails, NCP reports a Down event and enters the Link Termination phase.

   If the interface is configured with an IP address, the IPCP negotiation is performed. IPCP configuration options include IP addresses and DNS server IP addresses. After the IPCP negotiation succeeds, the link can carry IP packets.
5. After the NCP negotiation is performed, the PPP link remains active until either of the following events occurs:
   - Explicit LCP or NCP frames close the link.
   - Some external events take place (for example, the intervention of a user).

PPP authentication

PPP supports the following authentication methods:

PAP

PAP is a two-way handshake authentication protocol using the username and password.

PAP sends username/password pairs in plain text over the network. If authentication packets are intercepted in transit, network security might be threatened. For this reason, it is suitable only for low-security environments.

CHAP

CHAP is a three-way handshake authentication protocol.

CHAP transmits usernames but not passwords over the network. It transmits the result calculated from the password and random packet ID by using the MD5 algorithm. It is more secure than PAP. The authenticator may or may not be configured with a username. As a best practice, configure a username for the authenticator, which makes it easier for the peer to verify the identity of the authenticator.

MS-CHAP

MS-CHAP is a three-way handshake authentication protocol. MS-CHAP differs from CHAP as follows:

- MS-CHAP uses CHAP Algorithm 0x80.
- MS-CHAP provides authentication retry. If the peer fails authentication, it is allowed to retransmit authentication information to the authenticator for reauthentication. The authenticator allows a peer to retransmit a maximum of three times.

MS-CHAP-V2

MS-CHAP-V2 is a three-way handshake authentication protocol. MS-CHAP-V2 differs from CHAP as follows:

- MS-CHAP-V2 uses CHAP Algorithm 0x81.
- MS-CHAP-V2 provides two-way authentication by piggybacking a peer challenge on the Response packet and an authenticator response on the Acknowledge packet.
- MS-CHAP-V2 supports authentication retry. If the peer fails authentication, it is allowed to retransmit authentication information to the authenticator for reauthentication. The authenticator allows a peer to retransmit a maximum of three times.
- MS-CHAP-V2 supports password change. If the peer fails authentication because of an expired password, it will send the new password entered by the user to the authenticator for reauthentication.

PPP for IPv4

On IPv4 networks, PPP negotiates the IP address and DNS server address during IPCP negotiation.

IP address negotiation

IP address negotiation enables one end to assign an IP address to the other.

An interface can act as a client or a server during IP address negotiation:

- **Client**—Obtains an IP address from the server. Use the client mode when the device accesses the Internet through an ISP.
- **Server**—Assigns an IP address to the client. Before you configure the IP address of the server, you must perform one of the following tasks:
  - Configure a local address pool and associate the pool with the ISP domain.
  - Specify an IP address or an address pool for the client on the interface.

When IP address negotiation is enabled on a client, the server selects an IP address for the client in the following sequence:

1. If the AAA server configures an IP address or address pool for the client, the server selects that IP address or an IP address from the pool. The IP address or address pool is configured on the AAA server instead of the PPP server. For information about AAA, see *Security Configuration Guide*.
2. If an address pool is associated with the ISP domain used during client authentication, the server selects an IP address from the pool.
3. If an IP address or address pool is specified for the client on the interface of the server, the server selects that IP address or an IP address from that pool.

DNS server address negotiation

IPCP negotiation can determine the DNS server IP address.

When the device is connected to a host, configure the device as the server to assign the DNS server IP address to the host.

When the device is connected to an ISP access server, configure the device as the client. Then, the device can obtain the DNS server IP address from the ISP access server.

PPP for IPv6

On IPv6 networks, PPP negotiates only the IPv6 interface identifier instead of the IPv6 address and IPv6 DNS server address during IPv6CP negotiation.

IPv6 address assignment

PPP cannot negotiate the IPv6 address.

The client can get an IPv6 global unicast address through the following methods:

- **Method 1**—The client obtains an IPv6 prefix in an RA message. The client then generates an IPv6 global unicast address by combining the IPv6 prefix and the negotiated IPv6 interface identifier. The IPv6 prefix in the RA message is determined in the following sequence:
  - IPv6 prefix authorized by AAA.
  - RA prefix configured on the interface.
  - Prefix of the IPv6 global unicast address configured on the interface.

  For information about the ND protocol, see *Layer 3—IP Services Configuration Guide*.
- **Method 2**—The client requests an IPv6 global unicast address through DHCPv6. The server assigns an IPv6 address to the client from the address pool authorized by AAA. If no AAA-authorized address pool exists, DHCPv6 uses the address pool that matches the server's

IPv6 address to assign an IPv6 address to the client. For information about DHCPv6, see *Layer 3—IP Services Configuration Guide*.

- **Method 3**—The client requests prefixes through DHCPv6 and assigns them to downstream hosts. The hosts then uses the prefixes to generate global IPv6 addresses. This method uses the same principle of selecting address pools as method 2.

The device can assign a host an IPv6 address in either of the following ways:

- When the host connects to the device directly or through a bridge device, the device can use method 1 or method 2.
- When the host accesses the device through a router, the device can use method 3 to assign an IPv6 prefix to the router. The router assigns the prefix to the host to generate an IPv6 global unicast address.

IPv6 DNS server address assignment

On IPv6 networks, two methods are available for the IPv6 DNS address assignment:

- AAA authorizes the IPv6 DNS address and assigns this address to the host through RA messages.
- The DHCPv6 client requests an IPv6 DNS address from the DHCPv6 server.

PPP tasks at a glance

To configure PPP, perform the following tasks:

1. Configuring a VT interface
   - Creating a VT interface

   Perform this task in PPPoE and L2TP networks.
   - (Optional.) Restoring the default settings for the VT interface
2. Configuring PPP authentication

   Choose one of the following tasks:
   - Configuring PAP authentication
   - Configuring CHAP authentication (authenticator name is configured)
   - Configuring CHAP authentication (authenticator name is not configured)
   - Configuring MS-CHAP or MS-CHAP-V2 authentication

   Configure PPP authentication for high-security environments.
3. (Optional.) Configuring the polling feature
4. (Optional.) Configuring PPP negotiation
   - Configuring the PPP negotiation timeout time
   - Configuring IP address negotiation on the client
   - Configuring IP address negotiation on the server
   - Enabling IP segment match
   - Configuring DNS server IP address negotiation on the client
   - Configuring DNS server IP address negotiation on the server
   - Configuring ACFC negotiation
   - Configuring PFC negotiation
5. (Optional.) Enabling IP header compression

   IPHC is often used for voice communications over low-speed links.
6. (Optional.) Configuring the NAS-Port-Type attribute
7. (Optional.) Enabling PPP accounting

8. (Optional.) Enabling PPP user logging

Configuring a VT interface

Creating a VT interface

About this task

A virtual-template (VT) interface is a template for creating VA interfaces. In PPPoE, L2TP, and MP networks, VA interfaces are needed for exchanging data with peers. In this case, the system will select a VT interface and dynamically create VA interfaces based on the VT interface.

In PPPoE and L2TP applications, you can use VT interfaces to implement related functions of PPP. For more information about PPPoE and L2TP, see "Configuring PPPoE" and L2TP configuration in *VPN Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a VT interface and enter its view.

   interface virtual-template *number*

3. (Optional.) Set the interface description.

   description *text*

   By default, the description of a VT interface is *interface name* **Interface**, for example, **Virtual-Template1 Interface**.

4. (Optional.) Set the MTU size of the interface.

   mtu *size*

   By default, the MTU size of an interface is 1500 bytes.

5. (Optional.) Set the expected bandwidth of the VT interface.

   bandwidth *bandwidth-value*

   By default, the expected bandwidth (in kbps) is the interface baud rate divided by 1000.

Restoring the default settings for the VT interface

Restrictions and guidelines

The **default** command might interrupt ongoing network services. Make sure you are fully aware of the impact of this command when you execute it on a live network.

The **default** command might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **display this** command in interface view to identify these commands. Use the **undo** forms of these commands or follow the command reference to individually restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.

   system-view

2. Enter VT interface view.

   interface virtual-template *number*

3. Restore the default settings for the interface.

default

Configuring PPP authentication

About PPP authentication

PPP supports authentication methods PAP, CHAP, MS-CHAP, and MS-CHAP-V2. You can configure several authentication modes simultaneously. In LCP negotiation, the authenticator negotiates with the peer in the sequence of configured authentication modes until the LCP negotiation succeeds. If the response packet from the peer carries a recommended authentication mode, the authenticator directly uses the authentication mode if it finds the mode configured.

Configuring PAP authentication

Restrictions and guidelines for PAP authentication

For local AAA authentication, the username and password of the peer must be configured on the authenticator.

For remote AAA authentication, the username and password of the peer must be configured on the remote AAA server.

The username and password configured for the peer must be the same as those configured on the peer by using the **ppp pap local-user** command.

Configuring the authenticator

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the authenticator to authenticate the peer by using PAP.

   ppp authentication-mode pap [[**call-in**] **domain** { *isp-name* | **default enable** *isp-name* }]

   By default, PPP authentication is disabled.

4. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring the peer

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the PAP username and password sent from the peer to the authenticator when the peer is authenticated by the authenticator by using PAP.

   ppp pap local-user *username* **password** { **cipher** | **simple** } *string*

   By default, when being authenticated by the authenticator by using PAP, the peer sends null username and password to the authenticator.

   For security purposes, the password specified in plaintext form and ciphertext form will be stored in encrypted form.

Configuring CHAP authentication (authenticator name is configured)

Restrictions and guidelines for CHAP authentication (authenticator name is configured)

When you configure the authenticator, follow these guidelines:

- For local AAA authentication, the username and password of the peer must be configured on the authenticator.
- For remote AAA authentication, the username and password of the peer must be configured on the remote AAA server.
- The username and password configured for the peer must meet the following requirements:
  - The username configured for the peer must be the same as that configured on the peer by using the **`ppp chap user`** command.
  - The passwords configured for the authenticator and peer must be the same.

When you configure the peer, follow these guidelines:

- For local AAA authentication, the username and password of the authenticator must be configured on the peer.
- For remote AAA authentication, the username and password of the authenticator must be configured on the remote AAA server.
- The username and password configured for the authenticator must meet the following requirements:
  - The username configured for the authenticator must be the same as that configured on the authenticator by using the **`ppp chap user`** command.
  - The passwords configured for the authenticator and peer must be the same.
- The peer does not support the CHAP authentication password configured by using the **`ppp chap password`** command. CHAP authentication (authenticator name is configured) will apply even if the authentication name is configured.

Configuring the authenticator

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the authenticator to authenticate the peer by using CHAP.

   `ppp authentication-mode chap` [[**`call-in`**] **`domain`** { *`isp-name`* | **`default enable`** *`isp-name`* }]

   By default, PPP authentication is disabled.

4. Configure a username for the CHAP authenticator.

   `ppp chap user` *`username`*

   The default setting is null.

5. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring the peer

1. Enter system view.

   `system-view`

2. Enter interface view.

interface *interface-type interface-number*

3. Configure a username for the CHAP peer.

   ppp chap user *username*

   The default setting is null.

4. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring CHAP authentication (authenticator name is not configured)

Restrictions and guidelines for CHAP authentication (authenticator name is not configured)

For local AAA authentication, the username and password of the peer must be configured on the authenticator.

For remote AAA authentication, the username and password of the peer must be configured on the remote AAA server.

The username and password configured for the peer must meet the following requirements:

- The username configured for the peer must be the same as that configured on the peer by using the **ppp chap user** command.
- The password configured for the peer must be the same as that configured on the peer by using the **ppp chap password** command.

Configuring the authenticator

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the authenticator to authenticate the peer by using CHAP.

   ppp authentication-mode chap [[**call-in**] **domain** { *isp-name* | **default enable** *isp-name* }]

   By default, PPP authentication is disabled.

4. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring the peer

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a username for the CHAP peer.

   ppp chap user *username*

   The default setting is null.

4. Set the CHAP authentication password.

   ppp chap password { **cipher** | **simple** } *string*

   The default setting is null.

   For security purposes, the password specified in plaintext form and ciphertext form will be stored in encrypted form.

Configuring MS-CHAP or MS-CHAP-V2 authentication

Restrictions and guidelines for MS-CHAP or MS-CHAP-V2 authentication

The device can only act as an authenticator for MS-CHAP or MS-CHAP-V2 authentication.

MS-CHAP-V2 authentication supports password change only when using RADIUS.

As a best practice, do not set the authentication method for PPP users to **none** when MS-CHAP-V2 authentication is used.

For local AAA authentication, the username and password of the peer must be configured on the authenticator. For remote AAA authentication, the username and password of the peer must be configured on the remote AAA server. The username and password of the peer configured on the authenticator or remote AAA server must be the same as those configured on the peer.

If authentication name is configured, the username configured for the authenticator on the peer must be the same as that configured on the authenticator by using the **ppp chap user** command.

Configuring MS-CHAP or MS-CHAP-V2 authentication (authenticator name is configured)

1. Enter system view.

   system-view

2. Enter interface view.

   interface interface-type interface-number

3. Configure the authenticator to authenticate the peer by using MS-CHAP or MS-CHAP-V2.

   ppp authentication-mode { **ms-chap** | **ms-chap-v2** } [[**call-in**] **domain** { *isp-name* | **default enable** *isp-name* }]

   By default, PPP authentication is disabled.

4. Configure a username for the MS-CHAP or MS-CHAP-V2 authenticator.

   ppp chap user *username*

5. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring MS-CHAP or MS-CHAP-V2 authentication (authenticator name is not configured)

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the authenticator to authenticate the peer by using MS-CHAP or MS-CHAP-V2.

   ppp authentication-mode { **ms-chap** | **ms-chap-v2** } [[**call-in**] **domain** { *isp-name* | **default enable** *isp-name* }]

   By default, PPP authentication is disabled.

4. Configure local or remote AAA authentication.

   For more information about AAA authentication, see *Security Configuration Guide*.

Configuring the polling feature

About this task

The polling feature checks PPP link state.

On an interface that uses PPP encapsulation, the link layer sends keepalive packets at keepalive intervals to detect the availability of the peer. If the interface receives no response to keepalive packets when the keepalive retry limit is reached, it determines that the link fails and reports a link layer down event.

To set the keepalive retry limit, use the **timer-hold retry** command.

The value 0 disables an interface from sending keepalive packets. In this case, the interface can respond to keepalive packets from the peer.

Restrictions and guidelines

On a slow link, increase the keepalive interval to prevent false shutdown of the interface. This situation might occur when keepalive packets are delayed because a large packet is being transmitted on the link.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the keepalive interval.

   timer-hold *seconds*

   The default setting is 10 seconds.

4. Set the keepalive retry limit.

   timer-hold retry *retries*

   The default setting is 5.

Configuring PPP negotiation

Configuring the PPP negotiation timeout time

About this task

The device starts the PPP negotiation timeout timer after sending a packet. If no response is received before the timer expires, the device sends the packet again.

If two ends of a PPP link vary greatly in the LCP negotiation packet processing rate, configure this command on the end with a higher processing rate. The LCP negotiation delay timer prevents frequent LCP negotiation packet retransmission. After the physical layer comes up, PPP starts LCP negotiation when the delay timer expires. If PPP receives LCP negotiation packets before the delay timer expires, it starts LCP negotiation immediately.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the negotiation timeout time.

   ppp timer negotiate *seconds*

   The default setting is 3 seconds.

4. (Optional.) Set the LCP negotiation delay timer.

   ppp lcp delay *milliseconds*

   By default, PPP starts LCP negotiation immediately after the physical layer comes up.

Configuring IP address negotiation on the client

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable IP address negotiation.

   ```
   ip address ppp-negotiate
   ```

   By default, IP address negotiation is not enabled.

   If you execute this command and the **ip address** command multiple times, the most recent configuration takes effect. For more information about the **ip address** command, see *Layer 3—IP Services Command Reference*.

Configuring IP address negotiation on the server

About this task

Configure the server to assign an IP address to a client by using the following methods:

- Method 1: Specify an IP address for the client on the server interface.
- Method 2: Specify a PPP or DHCP address pool on the server interface.
- Method 3: Associate a PPP or DHCP address pool with an ISP domain.

Restrictions and guidelines for IP address negotiation on the server

For clients requiring no authentication, you can use either method 1 or method 2. When both method 1 and method 2 are configured, the most recent configuration takes effect.

For clients requiring authentication, you can use one or more of the three methods. When multiple methods are configured, method 3 takes precedence over method 1 or method 2. When both method 1 and method 2 are configured, the most recent configuration takes effect.

PPP supports IP address assignment from a PPP or DHCP address pool. If you use a pool name that identifies both a PPP address pool and a DHCP address pool, the system uses the PPP address pool.

When assigning IP address to users through a PPP address pool, make sure the PPP address pool excludes the gateway IP address of the PPP address pool.

Specifying an IP address for the client on the server interface

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Configure the interface to assign an IP address to the peer.

   ```
   remote address ip-address
   ```

   By default, an interface does not assign an IP address to the peer.

4. Configure an IP address for the interface.

   ```
   ip address ip-address
   ```

   By default, no IP address is configured on an interface.

Specifying a PPP address pool on the server interface

1. Enter system view.

system-view

2. Configure a PPP address pool.

   ip pool *pool-name start-ip-address* [*end-ip-address*] [**group** *group-name*]

3. (Optional.) Configure a gateway address for the PPP address pool.

   ip pool *pool-name* **gateway** *ip-address* [**vpn-instance** *vpn-instance-name*]

   By default, the PPP address pool is not configured with a gateway address.

4. (Optional.) Configure a PPP address pool route.

   ppp ip-pool route *ip-address* { *mask-length* | *mask* } [**vpn-instance** *vpn-instance-name*]

   By default, no PPP address pool route exists.

   The destination network of the PPP address pool route must include the PPP address pool.

5. Enter interface view.

   interface *interface-type interface-number*

6. Configure the interface to assign an IP address from the configured PPP address pool to the peer.

   remote address pool *pool-name*

   By default, an interface does not assign an IP address to the peer.

7. Configure an IP address for the interface.

   ip address *ip-address*

   By default, no IP address is configured on an interface.

Specifying a DHCP address pool on the server interface

1. Enter system view.

   system-view

2. Configure DHCP.
   - If the server acts as a DHCP server, perform the following tasks:
     - Configure the DHCP server.
     - Configure a DHCP address pool on the server.
   - If the server acts as a DHCP relay agent, perform the following tasks:
     - Configure the DHCP relay agent on the server.
     - Configure a DHCP address pool on the remote DHCP server.
     - Enable the DHCP relay agent to record relay entries.
     - Configure a DHCP relay address pool.

   For information about configuring a DHCP server and a DHCP relay agent, see *Layer 3—IP Services Configuration Guide*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Configure the interface to assign an IP address from the configured DHCP address pool to the peer.

   remote address pool *pool-name*

   By default, an interface does not assign an IP address to the peer.

5. (Optional.) Configure the DHCP client IDs for PPP users acting as DHCP clients.

   remote address dhcp client-identifier { **callingnum** | **username** }

By default, no DHCP client IDs are configured for PPP users acting as DHCP clients.

When PPP usernames are used as DHCP client IDs, make sure different users use different PPP usernames to come online.

6. Configure an IP address for the interface.

   ip address *ip-address*

   By default, no IP address is configured on an interface.

Associating a PPP address pool with an ISP domain

1. Enter system view.

   system-view

2. Configure a PPP address pool.

   ip pool *pool-name start-ip-address* [*end-ip-address*] [**group** *group-name*]

   By default, no PPP address pool is configured.

3. (Optional.) Configure a gateway address for the PPP address pool.

   ip pool *pool-name* **gateway** *ip-address* [**vpn-instance** *vpn-instance-name*]

   By default, the PPP address pool is not configured with a gateway address.

4. (Optional.) Configure a PPP address pool route.

   ppp ip-pool route *ip-address* { *mask-length* | *mask* } [**vpn-instance** *vpn-instance-name*]

   By default, no PPP address pool route exists.

   The destination network of the PPP address pool route must include the PPP address pool.

5. Enter ISP domain view.

   domain *isp-name*

6. Associate the ISP domain with the configured PPP address pool for address assignment.

   authorization-attribute ip-pool *pool-name*

   By default, no PPP address pool is associated.

   For more information about this command, see *Security Command Reference.*

7. Return to system view.

   quit

8. Enter interface view.

   interface *interface-type interface-number*

9. Configure an IP address for the interface.

   ip address *ip-address*

   By default, no IP address is configured on an interface.

Associating a DHCP address pool with an ISP domain

1. Enter system view.

   system-view

2. Configure DHCP.

   - If the server acts as a DHCP server, perform the following tasks:
     - Configure the DHCP server.
     - Configure a DHCP address pool on the server.
   - If the server acts as a DHCP relay agent, perform the following tasks:
     - Configure the DHCP relay agent on the server.

- Configure a DHCP address pool on the remote DHCP server.
- Enable the DHCP relay agent to record relay entries.
- Configure a DHCP relay address pool.

For information about configuring a DHCP server and a DHCP relay agent, see *Layer 3—IP Services Configuration Guide*.

3. Enter ISP domain view.

   domain *isp-name*

4. Associate the ISP domain with the configured DHCP address pool or DHCP relay address pool for address assignment.

   authorization-attribute ip-pool *pool-name*

   By default, no DHCP address pool or DHCP relay address pool is associated.

   For more information about this command, see *Security Command Reference*.

5. Return to system view.

   quit

6. Enter interface view.

   interface *interface-type interface-number*

7. (Optional.) Configure the DHCP client IDs for PPP users acting as DHCP clients.

   remote address dhcp client-identifier { **callingnum** | **username** }

   By default, no DHCP client IDs are configured for PPP users acting as DHCP clients.

   When PPP usernames are used as DHCP client IDs, make sure different users use different PPP usernames to come online.

8. Configure an IP address for the interface.

   ip address *ip-address*

   By default, no IP address is configured on an interface.

Enabling IP segment match

About this task

This feature enables the local interface to check whether its IP address and the IP address of the remote interface are in the same network segment. If they are not, IPCP negotiation fails.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IP segment match.

   ppp ipcp remote-address match

   By default, this feature is disabled.

Configuring DNS server IP address negotiation on the client

About this task

During PPP negotiation, the server will assign a DNS server IP address only for a client configured with the **ppp ipcp dns request** command. For some special devices to forcibly assign DNS server IP addresses to clients that do not initiate requests, configure the **ppp ipcp dns admit-any** command on these devices.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the device to request the peer for a DNS server IP address.

   ppp ipcp dns request

   By default, a client does not request its peer for a DNS server IP address.

4. Configure the device to accept the DNS server IP addresses assigned by the peer even though it does not request the peer for the DNS server IP addresses.

   ppp ipcp dns admit-any

   By default, a device does not accept the DNS server IP addresses assigned by the peer if it does not request the peer for the DNS server IP addresses.

   This command is not necessary if the **ppp ipcp dns request** command is configured.

Configuring DNS server IP address negotiation on the server

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the primary and secondary DNS server IP addresses to be allocated to the peer in PPP negotiation.

   ppp ipcp dns *primary-dns-address* [*secondary-dns-address*]

   By default, a device does not allocate DNS server IP addresses to its peer if the peer does not request them.

   After this command is configured, the server allocate DNS server IP addresses to a client that initiates requests.

Configuring ACFC negotiation

About this task

PPP can compress the address and control fields of PPP packets to increase the payload size.

ACFC negotiation notifies the peer that the local end can receive packets carrying compressed address and control fields.

ACFC negotiation is implemented at the LCP negotiation stage. After the ACFC negotiation succeeds, PPP does not include the address and control fields in non-LCP packets. To ensure successful LCP negotiation, PPP does not apply the compression to LCP packets.

Restrictions and guidelines for ACFC negotiation

As a best practice, use the ACFC configuration option on low-speed links.

Configuring the local end to send ACFC requests

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the local end to send ACFC requests by including the ACFC option in outbound LCP negotiation requests.

   ppp acfc local request

   By default, the local end does not include the ACFC option in outbound LCP negotiation requests.

Configuring local end to reject ACFC requests received from the peer

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the local end to reject ACFC requests received from the peer.

   ppp acfc remote-reject

   By default, the local end accepts the ACFC requests from the remote peer, and performs ACFC on frames sent to the peer.

Configuring PFC negotiation

About this task

PPP can compress the protocol field of PPP packets from 2 bytes to 1 byte to increase the payload size.

PFC negotiation notifies the peer that the local end can receive packets with a single-byte protocol field.

PFC negotiation is implemented at the LCP negotiation stage. After PFC negotiation is completed, the device compresses the protocol field of sent non-LCP packets. If the first eight bits of the protocol field are all zeros, the device does not add those bits into the packet. To ensure successful LCP negotiation, PPP does not apply the compression to LCP packets.

Restrictions and guidelines for PFC negotiation

As a best practice, use this configuration option on low-speed links.

Configuring the local end to send PFC requests

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the local end to send PFC requests by including the PFC option in outbound LCP negotiation requests.

   ppp pfc local request

   By default, the local end does not include the PFC option in outbound LCP negotiation requests.

Configuring the local end to reject PFC requests received from the peer

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the local end to reject PFC requests received from the peer.

   ppp pfc remote-reject

   By default, the device accepts PFC requests received from the peer, and performs PFC on frames sent to the peer.

Enabling IP header compression

About this task

IP header compression (IPHC) compresses packet headers to speed up packet transmission. IPHC is often used for voice communications over low-speed links.

IPHC provides the following compression features:

- **RTP header compression**—Compresses the IP header, UDP header, and RTP header of an RTP packet, which have a total length of 40 bytes.
- **TCP header compression**—Compresses the IP header and TCP header of a TCP packet, which have a total length of 40 bytes.

Restrictions and guidelines

To use IPHC, you must enable it on both sides of a PPP link.

Enabling or disabling IPHC on a VT or dialer interface does not immediately take effect. You must execute the **shutdown** and **undo shutdown** commands on the interface or the bound physical interface to apply the new setting.

After you enable IPHC, you can configure the maximum number of connections for RTP or TCP header compression. The configuration takes effect after you execute the **shutdown** and **undo shutdown** command on the interface. The configuration is removed after IPHC is disabled.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IP header compression.

   ppp compression iphc enable [**nonstandard**]

   By default, IP header compression is disabled.

   The **nonstandard** option must be specified when the device communicates with a non-NSFOCUS device.

   When the **nonstandard** keyword is specified, only RTP header compression is supported and TCP header compression is not supported.

4. Set the maximum number of connections for which an interface can perform RTP header compression.

   ppp compression iphc rtp-connections *number*

   The default setting is 16.

5. Set the maximum number of connections for which an interface can perform TCP header compression.

   ppp compression iphc tcp-connections *number*

   The default setting is 16.

Configuring the NAS-Port-Type attribute

About this task

The NAS-Port-Type attribute is used for RADIUS authentication and accounting. For information about the NAS-Port-Type attribute, see RFC 2865.

Restrictions and guidelines

The configuration of this feature does not affect existing users.

Procedure

1. Enter system view.

   `system-view`

2. Enter VT interface view.

   `interface virtual-template` *`number`*

3. Configure the NAS-Port-Type attribute.

   `nas-port-type` { **`ethernet`** | **`virtual`** }

   By default, the NAS-Port-Type attribute is determined by the service type and link type of the PPP user as follows:

   - When the service type is PPPoE, the NAS-Port-Type attribute is **`ethernet`**.
   - When the service type is L2TP, the NAS-Port-Type attribute is **`virtual`**.

Enabling PPP accounting

About this task

PPP accounting collects PPP statistics, including the numbers of received and sent PPP packets and bytes. AAA can use the PPP statistics for accounting. For more information about AAA, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable PPP accounting.

   `ppp account-statistics enable` [**`acl`** { *`acl-number`* | **`name`** *`acl-name`* }]

   By default, PPP accounting is disabled.

Enabling PPP user logging

About this task

The PPP user logging feature enables the device to generate PPP logs and send them to the information center. Logs are generated after a user comes online, goes offline, or fails to come online. A log entry contains information such as the username, IP address, interface name, inner VLAN, outer VLAN, MAC address, and failure causes. For information about the log destination and output rule configuration in the information center, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

Typically, disable this feature to prevent excessive PPP log output.

Procedure

1. Enter system view.
   `system-view`
2. Enable PPP user logging.
   `ppp access-user log enable` [**`successful-login`** | **`failed-login`** | **`normal-logout`** | **`abnormal-logout`**] *
   By default, PPP user logging is disabled.

Display and maintenance commands for PPP

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display information about VA interfaces.	**display interface** [**virtual-access** [*interface-number*]] [**brief** [**description** | **down**]]
Display information about VT interfaces.	**display interface** [**virtual-template** [*interface-number*]] [**brief** [**description** | **down**]]
Display PPP address pools.	**display ip pool** [*pool-name* | **group** *group-name*]
Display information about PPP access users.	**display ppp access-user** { **domain** *domain-name* | **interface** *interface-type interface-number* [**count**] | **ip-address** *ipv4-address* | **ipv6-address** *ipv6-address* | **username** *user-name* | **user-type** { **lac** | **lns** | **pppoe** } [**count**] }
Display PPP negotiation packet statistics.	**display ppp packet statistics** [**slot** *slot-number*]
Display IPHC statistics.	**display ppp compression iphc** { **rtp** | **tcp** } [**interface** *interface-type interface-number*]
Clear statistics on VA interfaces.	**reset counters interface** [**virtual-access** [*interface-number*]]
Log off a PPP user.	**reset ppp access-user** { **ip-address** *ipv4-address* [**vpn-instance** *ipv4-vpn-instance-name*] | **ipv6-address** *ipv6-address* [**vpn-instance** *ipv6-vpn-instance-name*] | **username** *user-name* }
Clear IPHC statistics.	**reset ppp compression iphc** [**rtp** | **tcp**] [**interface** *interface-type interface-number*]
Clear PPP negotiation packet statistics.	**reset ppp packet statistics** [**slot**

Task	Command
	slot-number]

Configuring PPPoE

About PPPoE

Point-to-Point Protocol over Ethernet (PPPoE) extends PPP by transporting PPP frames encapsulated in Ethernet over point-to-point links.

PPPoE specifies the methods for establishing PPPoE sessions and encapsulating PPP frames over Ethernet. PPPoE requires a point-to-point relationship between peers instead of a point-to-multipoint relationship as in multi-access environments such as Ethernet. PPPoE provides Internet access for the hosts in an Ethernet through a remote access device and implement access control, authentication, and accounting on a per-host basis. Integrating the low cost of Ethernet and scalability and management functions of PPP, PPPoE gained popularity in various application environments, such as residential access networks.

For more information about PPPoE, see RFC 2516.

PPPoE network structure

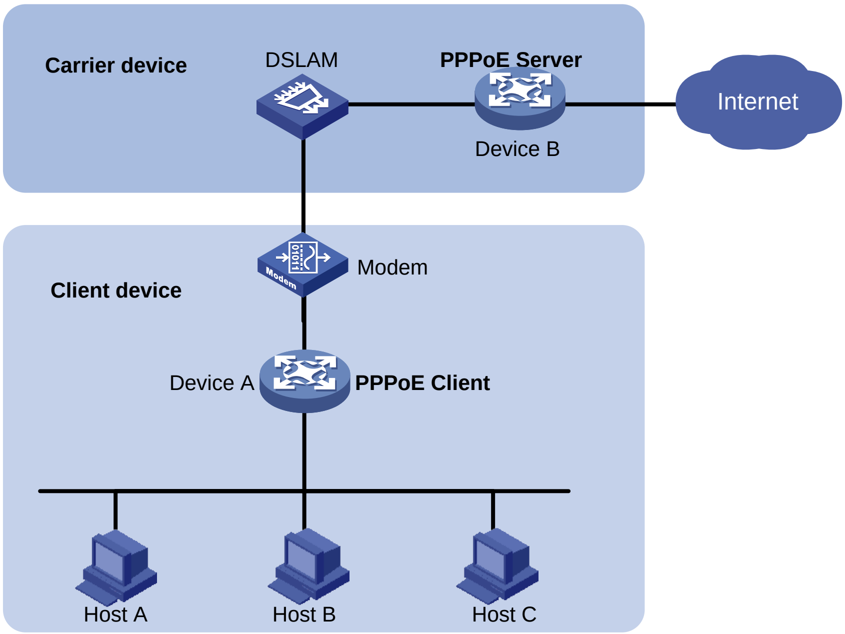
PPPoE uses the client/server model. The PPPoE client initiates a connection request to the PPPoE server. After session negotiation between them is complete, a session is established between them, and the PPPoE server provides access control, authentication, and accounting to the PPPoE client.

PPPoE network structures are classified into router-initiated and host-initiated network structures depending on the starting point of the PPPoE session.

Router-initiated network structure

As shown in Figure 2, the PPPoE session is established between devices (Device A and Device B). All hosts share one PPPoE session for data transmission without being installed with PPPoE client software. This network structure is typically used by enterprises.

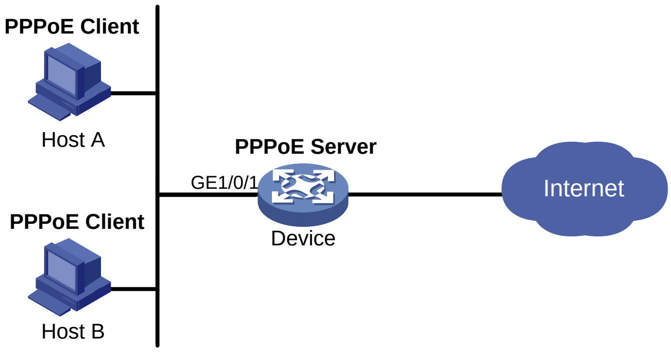
Figure 2 Router-initiated network structure



Host-initiated network structure

As shown in Figure 3, a PPPoE session is established between each host (PPPoE client) and the carrier device (PPPoE server). The service provider assigns an account to each host for billing and control. The host must be installed with PPPoE client software.

Figure 3 Host-initiated network structure



Configuring a PPPoE client

Operation mode

A PPPoE session can operate in one of the following modes:

- **Permanent mode**—A PPPoE session is established immediately when the line is physically up. This type of session remains until the physical link comes down or until the session is disconnected.
- **On-demand mode**—A PPPoE session is established when there is a demand for data transmission instead of when the line is physically up. It is terminated when idled for a specific period of time.
- **Diagnostic mode**—A PPPoE session is established immediately after the device configurations finish. The device automatically terminates the PPPoE session and then tries to re-establish a PPPoE session at a pre-configured interval. By establishing and terminating PPPoE sessions periodically, you can monitor the operating status of the PPPoE link.

The PPPoE session operating mode is determined by your configuration on the dialer interface:

- **Permanent mode**—Used when you set the link idle time to 0 by using the **`dialer timer idle`** command and do not configure the **`dialer diagnose`** command.
- **On-demand mode**—Used when you set the link idle time to a non-zero value by using the **`dialer timer idle`** command and do not configure the **`dialer diagnose`** command.
- **Diagnostic mode**—Used when you configure the **`dialer diagnose`** command.

PPPoE client tasks at a glance

To configure a PPPoE client, perform the following tasks:

1. Configuring a dialer interface
2. Configuring a PPPoE session
3. (Optional.) Resetting a PPPoE session

Configuring a dialer interface

About this task

Before establishing a PPPoE session, you must first create a dialer interface and configure bundle DDR on the interface. Each PPPoE session uniquely corresponds to a dialer bundle, and each dialer bundle uniquely corresponds to a dialer interface. A PPPoE session uniquely corresponds to a dialer interface.

Procedure

1. Enter system view.

   system-view

2. Create a dialer group and configure a dial rule.

   dialer-group *group-number* **rule** { **ip** | **ipv6** } { **deny** | **permit** | **acl** { *acl-number* | **name** *acl-name* } }

   Configure this command only when the PPPoE session operates in on-demand mode.

3. Create a dialer interface and enter its view.

   interface dialer *number*

4. Assign an IP address to the interface.

   ip address { *address mask* | **ppp-negotiate** }

   By default, no IP address is configured.

5. Enable bundle DDR on the interface.

   dialer bundle enable

   By default, bundle DDR is disabled.

6. Associate the interface with the dial rule by associating the interface with the corresponding dialer group.

   dialer-group *group-number*

   By default, a dialer interface is not assigned to any dialer group.

   Configure this command only when the PPPoE session operates in on-demand mode.

7. Configure the link-idle timeout timer.

   dialer timer idle *idle* [**in** | **in-out**]

   The default setting is 120 seconds.

   When this timer is set to 0 seconds, the PPPoE session operates in permanent mode. Otherwise, the PPPoE session operates in on-demand mode.

8. Configure the DDR application to operate in diagnostic mode.

   dialer diagnose [**interval** *interval*]

   By default, the DDR application operates in non-diagnostic mode.

   Execute this command only when the PPPoE session operates in diagnostic mode.

9. (Optional.) Set the auto-dial interval.

   dialer timer autodial *autodial-interval*

   The default setting is 300 seconds.

   DDR starts the auto-dial timer after the link is disconnected and originates a new call when the auto-dial timer expires.

   As a best practice, set a shorter auto-dial interval for DDR to soon originate a new call.

10. (Optional.) Set the MTU for the dialer interface

    mtu *size*

    By default, the MTU on a dialer interface is 1500 bytes.

The dialer interface fragments a packet that exceeds the configured MTU, and adds a 2-byte PPP header and a 6-byte PPPoE header to each fragment. You should modify the MTU of a dialer interface to make sure the total length of any fragment packet is less than the MTU of the physical interface.

Configuring a PPPoE session

About this task

After a PPPoE session is successfully established, the system automatically creates a VA interface for exchanging packets with the peer. To display information about VA interfaces, execute the **display interface virtual-access** command. VA interfaces cannot be manually configured.

After the PPPoE session is terminated, the corresponding VA interface is automatically deleted.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Create a PPPoE session and specify a dialer bundle for the session.
   pppoe-client dial-bundle-number *number* [**no-hostuniq**]
   The *number* argument in this command must take the same value as the configured dialer interface number.

Resetting a PPPoE session

About this task

After you reset a PPPoE session in permanent mode, the device establishes a new PPPoE session when the autodial timer expires.

After you reset a PPPoE session in on-demand mode, the device establishes a new PPPoE session when there is a demand for data transmission.

Procedure

To reset a PPPoE session, execute the following command in user view:

reset pppoe-client { **all** | **dial-bundle-number** *number* }

Display and maintenance commands for PPPoE

Display and maintenance commands for PPPoE client

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display summary information for a PPPoE session.	**display pppoe-client session summary** [**dial-bundle-number** *number*]
Display the protocol packet statistics for a PPPoE session.	**display pppoe-client session packet** [**dial-bundle-number** *number*]

Task	Command
Clear the protocol packet statistics for a PPPoE session.	**reset pppoe-client session packet** [**dial-bundle-number** *number*]

Contents

Managing a mobile communication modem

About managing a mobile communication modem

A mobile communication modem connects a device to a mobile communication network.

USB 3G/4G modems and built-in modems are available for the device.

The device uses a fixed cellular interface to configure and manage a USB 3G/4G modem or a built-in modem. A USB 3G/4G modem is hot swappable. The configuration for a USB 3G/4G modem remains after the modem is removed from the device.

A cellular interface can be channelized into an Eth-channel interface. The data link layer protocol of the Eth-channel interface is Ethernet. The interface supports IP at the network layer.

The cellular interface of a 4G modem can only be channelized into an Eth-channel interface.

Restrictions: Hardware compatibility with mobile communication modem management

Model	Mobile communication modem management compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No
NFNX3-HDB680, NFNX3-HDB1080	Yes

Restrictions and guidelines: Mobile communication modem management

Follow these restrictions and guidelines when you configure mobile communication modem management:

- Do not remove a USB 3G/4G modem while it is transmitting data. As a best practice, execute the **shutdown** command to shut down the USB 3G/4G modem before removing it.
- A USB 3G/4G modem is not usable when the USB interface to which the modem is attached is shut down. For more information about shutting down a USB interface, see *Fundamentals Configuration Guide*.
- Unless otherwise noted, the mobile communication modem configuration in this document is saved in the NVM of the mobile communication modem. To verify the configuration, use the **display cellular** command.

Mobile communication modem management tasks at a glance

To management a mobile communication modem, perform the following tasks:

1. Configuring the mobile communication modem cellular interface
   - Configuring the 4G modem cellular interface
2. Configuring an Eth-channel interface for a 4G modem
3. Configuring a mobile communication network
4. Configuring parameter profiles
5. (Optional.) Specifying the primary or secondary SIM card
6. (Optional.) Associating mobile communication link backup with a track entry
7. (Optional.) Configuring PIN verification
8. (Optional.) Configuring DM
9. (Optional.) Setting the RSSI thresholds
10. (Optional.) Issuing a configuration directive to a mobile communication modem
11. (Optional.) Configuring mobile communication modem reboot
    - Configuring automatic reboot
    - Configuring manual reboot

Configuring the 4G modem cellular interface

1. Enter system view.

   `system-view`

2. Enter cellular interface view.

   `controller cellular` *`cellular-number`*

3. Configure a description for the cellular interface.

   `description` *`text`*

   By default, the description for a cellular interface is *interface name* **Interface**, for example, Cellular 1/0/1 Interface.

4. Channelize the cellular interface into an Eth-channel interface.

   `eth-channel` *`channel-number`*

   This command names the Eth-channel interface as eth-channel *cellular-number*:*channel-number*.

   Specify an IP address for the Eth-channel interface channelized from the cellular interface of a 4G modem as needed.

5. Bring up the cellular interface.

   `undo shutdown`

   By default, a cellular interface is up.

Configuring an Eth-channel interface for a 4G modem

Configuring basic parameters for an Eth-channel interface

1. Enter system view.

   system-view

2. Enter Eth-channel interface view.

   interface eth-channel *interface-number*

3. Configure a description of the Eth-channel interface.

   description *text*

   By default, the description for an Eth-channel interface is *interface name* **Interface**, for example, Echannel1/0/1:0 Interface.

4. Set the MTU for the Eth-channel interface.

   mtu *size*

   By default, the MTU for an Eth-channel interface is 1500 bytes.

5. Set the expected bandwidth of the Eth-channel interface.

   bandwidth *bandwidth-value*

   By default, the expected bandwidth (in kbps) of an Eth-channel is the interface baud rate divided by 1000.

   The baud rate of an Eth-channel interface is 100 Mbps.

6. Bring up the Eth-channel interface.

   undo shutdown

   By default, an Eth-channel interface is up.

Restoring the default settings for an Eth-channel interface

Restrictions and guidelines

△ **CAUTION:**

Restoring the default interface settings might interrupt ongoing network services. Make sure you are fully aware of the impact of this command when you use it on a live network.

The **default** command might fail to restore the default settings for some commands for reasons such as command dependencies and system restrictions. Use the **display this** command in interface view to identify these commands. Then use their **undo** forms or follow the command reference to individually restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.

   system-view

2. Enter Eth-channel interface view.

   interface eth-channel *interface-number*

3. Restore the default configurations.

   default

Configuring an IP address for an Eth-channel interface of a 4G modem

About this task

An Eth-channel interface can communicate with other devices only after it obtains an IP address. You can configure an IP address for an Eth-channel interface in the following ways:

- **DHCP**—The Eth-channel interface obtains the modem's IP address as its own IP address through DHCP. The modem's IP address is automatically allocated by the service provider.
- **Modem manufacturer's proprietary protocol**—The Eth-channel interface obtains the modem's IP address as its own IP address through the modem manufacturer's proprietary protocol. The modem's IP address is automatically allocated by the service provider.
- **Manual configuration**.

The ways for the Eth-channel interface to obtain an IP address are mutually exclusive. The most recent configuration overrides the previous one.

Restrictions and guidelines

Changing the IP address will result in dialup interruption. Immediate re-dialup upon interruption might not be supported by service providers.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter Eth-channel interface view.

   ```
   interface eth-channel interface-number
   ```

3. Configure an IP address for the Eth-channel interface in one of the three ways.

   - Obtain an IP address by using DHCP.

     ```
     ip address dhcp-alloc
     ```

     For more information about the command, see DHCP commands in *Layer 3–IP Service Command Reference*.

     By default, an Eth-channel interface does not obtain an IP address through the modem-manufacturer's proprietary protocol.

     For the support of hardware platforms for this command, see 3G and 4G modem commands in *Network Management and Monitoring Command Reference*.

   - Obtain an IP address by using the modem-manufacturer's proprietary protocol.

     IPv4:

     ```
     ip address cellular-alloc
     ```

     IPv6:

     ```
     ipv6 address cellular-alloc
     ```

     By default, an Eth-channel interface does not obtain an IP address through the modem-manufacturer's proprietary protocol.

   - Configure an IP address manually.

     ```
     ip address ip-address { mask-length | mask } [ sub ]
     ```

     By default, no IP address is specified for an Eth-channel interface.

Configuring a mobile communication network

Restrictions and guidelines

A 3G modem can access GSM, CDMA2000, TD-SCDMA, and WCDMA networks. A 4G modem can access GSM, CDMA2000, TD-SCDMA, WCDMA, and LTE networks.

A mobile communication modem is used to search a public land mobile network (PLMN) for accessible mobile networks. A PLMN is uniquely identified by the mobile country code (MCC) and the mobile network code (MNC). Some mobile communication modems can automatically access a mobile network. To manually specify a mobile network for a mobile communication modem, first search for available mobile networks.

Procedure

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. (Optional.) Search for PLMNs.

   plmn search

4. Configure a PLMN searching mode.

   plmn select { **auto** | **manual** *mcc mnc* }

   The default setting depends on the modem model.

5. Specify the network service to which the modem connects.

   mode { **1xrtt** | **auto** | **evdo** | **gsm** | **gsm-precedence** | **hybrid** | **lte** | **td** | **td-precedence** | **wcdma** | **wcdma-precedence** }

   The default setting for this command and support for the arguments depend on the modem model.

6. (Optional.) Specify a GSM frequency band.

   gsm band { **egsm900** | **gsm450** | **gsm480** | **gsm750** | **gsm850** | **gsm1800** | **gsm1900** | **pgsm900** | **rsgm900** }

   By default, no GSM frequency band is specified.

   This command is supported only by Sierra MC7354 (ATT version) and MC7304 4G modules.

7. (Optional.) Specify a WCDMA frequency band.

   wcdma band { **wcdma800** | **wcdma850** | **wcdma900** | **wcdma1700ip** | **wcdma1700us** | **wcdma1800** | **wcdma1900** | **wcdma2100** | **wcdma2600** }

   By default, no WCDMA frequency band is specified.

   This command is supported only by Sierra MC7354 (ATT version) and MC7304 4G modules.

8. (Optional.) Specify an LTE frequency band.

   lte band *band-number*

   The default setting for this command and support for the *band-number* argument depend on the 4G modem model.

Configuring parameter profiles

About parameter profiles

A parameter profile defines the following items:

- The access point with which a mobile communication modem is associated.
- The authentication mode in which the service provider authenticates the mobile communication modem.

Creating a parameter profile for a 3G modem

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. Create a parameter profile.

   profile create *profile-number* { **dynamic** | **static** *apn* } **authentication-mode** { **none** | { **chap** | **pap** } **user** *username* [**password** *password*] }

   The default setting depends on the modem model.

Creating a parameter profile for a 4G modem

1. Enter system view.

   system-view

2. Create a parameter profile and enter its view.

   apn-profile *profile-name*

3. Specify the PDP data carrying protocol.

   pdp-type { **ipv4** | **ipv6** | **ipv4v6** }

   By default, the data carrying protocol is IPv4 and IPv6.

4. Specify an APN for accessing a 4G network.

   apn { **dynamic** | **static** *apn* }

   By default, no APN is specified for accessing a 4G network.

5. Specify an authentication mode for accessing a 4G network.

   authentication-mode { **pap** | **chap** | **pap-chap** } **user** *user-name* **password** { **cipher** | **simple** } *string*

   By default, no authentication is performed for accessing a 4G network.

6. Specify a separator for the IMSI/SN binding authentication information.

   attach-format imsi-sn split *splitchart*

   By default, no separator is specified for the IMSI/SN binding authentication information.

Specifying the primary and backup profiles

About this task

By default, profile 1 is used for mobile communication modem dialup. The dialup fails if profile 1 does not exist.

You can also specify the primary and backup profiles for mobile communication modem dialup. The primary profile always has priority over the backup profile. For each dialup connection establishment, the mobile communication modem uses the backup profile only when it has failed to dial up using the primary profile.

Restrictions and guidelines

You must configure the same username and password for the primary and backup profiles.

Specifying the primary and backup profiles for a 3G modem

1. Enter system view.
   system-view
2. Enter cellular interface view.
   controller cellular *interface-number*
3. Specify the primary and backup profiles for the 3G modem.
   profile main *main-profile-number* **backup** *backup-profile-number*
   By default, profile 1 is used for 3G modem dialup.

Specifying the primary and backup profiles for a 4G modem

1. Enter system view.
   system-view
2. Enter Eth-channel interface view.
   interface eth-channel *interface-number*
3. Specify the primary and backup profiles for the 4G modem.
   apn-profile apply *profile-name* [**backup** *profile-name*]
   By default, no profiles are specified for 4G modem dialup.

Specifying the primary or secondary SIM card

About this task

On a device that uses dual SIM card, you can specify the use of the secondary SIM card when one of the following problems occurs:

- The mobile communication link signals of the primary SIM card are weak.
- The service provider network that the primary SIM card connects to is unavailable.
- The primary SIM card has failed.

If the problems of the primary SIM card are resolved, you can specify the use of the primary SIM card or enable the mobile communication modem to automatically switch back to the primary SIM card.

Procedure

1. Enter system view.
   system-view
2. Enter cellular interface view.
   controller cellular *interface-number*
3. Specify the primary or secondary SIM card.
   sim switch-to *card-number*
   By default, a mobile communication modem uses the primary SIM card.

Associating mobile communication link backup with a track entry

About this task

This configuration allows the system to use a track entry to monitor the status of the primary mobile communication link. When the track entry state changes from Positive to Negative, the secondary mobile communication link takes over.

Procedure

1. Enter system view.

   `system-view`

2. Enter cellular interface view.

   `controller cellular` *`interface-number`*

3. Associate mobile communication link backup with a track entry

   `sim backup enable track` *`entry-number`*

   By default, mobile communication link backup is not associated with a track entry.

Configuring PIN verification

About this task

A UIM card is used in the CDMA network and a SIM card is used in other mobile networks. Each SIM/UIM card has a Personal Identification Number (PIN). PIN verification prevents unauthorized access to the SIM/UIM card.

To perform PIN verification, you need to save the PIN of the SIM/UIM card on the device by using the **`pin verify`** command and enable PIN verification. The **`pin verify`** command can be executed before and after you enable PIN verification. After the PIN is saved on the device, the PIN is used for verification automatically when required.

When PIN verification is enabled, PIN verification is performed after you perform any of the following tasks:

- Install a mobile communication modem.
- Reboot the device where a USB mobile communication modem is attached.
- Execute the **`modem reboot`** command to reboot a mobile communication modem.
- Hot swap a USB mobile communication modem.

If PIN verification fails after a maximum number of attempts, the SIM/UIM card is locked, and a PIN Unlocking Key (PUK) is required to unlock the card. The maximum number of attempts depends on the mobile communication modem model.

Procedure

1. Enter system view.

   `system-view`

2. Enter cellular interface view.

   `controller cellular` *`interface-number`*

3. Enable PIN verification.

   `pin verification enable` [*`pin`*]

   The default setting depends on the modem model.

Whether you are required to enter the PIN when enabling PIN verification depends on the modem model.

4. Specify the PIN for verification.

   pin verify { **cipher** | **simple** } *string*

   By default, no PIN is configured on a device for mobile communication modem verification.

   This command saves the PIN on the device.

5. (Optional.) Specify a PUK to unlock the SIM/UIM card.

   pin unlock *puk new-pin*

   After the SIM/UIM card is unlocked, execute the **pin verify** command to save the new PIN on the device.

6. (Optional.) Modify the PIN of the SIM/UIM card.

   pin modify *current-pin new-pin*

   The new PIN is saved in the SIM/UIM card.

   After the PIN is modified, execute the **pin verify** command to save the new PIN on the device.

Configuring DM

NOTE:

Support for DM depends on the modem model.

About this task

Diagnostic and monitoring (DM) allows third-party debugging tools to diagnose and monitor the mobile communication modem through cellular interface debugging output. For more information about DM, see related mobile communication modem user manuals.

Procedure

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. Enable DM.

   dm-port open

   The default setting for this command depends on the modem model.

Setting the RSSI thresholds

About this task

After setting the RSSI thresholds, you will stay informed about the RSSI changes.

Procedure

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. Set the RSSI thresholds.

rssi { **gsm** | **1xrtt** | **evdo** | **lte** } { **low** *lowthreshold* | **medium** *mediumthreshold* }

By default, the lower and upper RSSI thresholds for a mobile communication modem are –150 dBm and 0 dBm, respectively.

The value of *lowthreshold* cannot be smaller than the value of *mediumthreshold* because the system automatically adds a negative sign to the RSSI thresholds.

Issuing a configuration directive to a mobile communication modem

Restrictions and guidelines

Configuration directives might cause malfunction of a mobile communication modem. When you issue a configuration directive to the modem, make sure you understand the impact on the mobile communication modem.

Procedure

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. Issue a configuration directive to the mobile communication modem.

   sendat *at-string*

Configuring mobile communication modem reboot

Configuring automatic reboot

About this task

A mobile communication modem might malfunction in an unstable mobile communication network or when the application environment changes. During a malfunction, the modem cannot respond to the device's requests or configuration commands. If the device does not receive any responses from the mobile communication modem within the timeout interval, a response failure occurs. When the number of consecutive response failures reaches the threshold, the device restarts the mobile communication modem automatically.

The device does not restart the mobile communication modem when the mobile communication modem has not made a successful dialup since the last restart. This restriction avoids repeated restarts of the mobile communication modem when there are configuration errors.

Procedure

1. Enter system view.

   system-view

2. Enter cellular interface view.

   controller cellular *interface-number*

3. Configure the response timeout and the threshold for the number of consecutive timeouts to enable automatic reboot of the mobile communication modem.

   modem response timer *time* **auto-recovery** *threshold*

   By default, the response timeout is 10 seconds and the consecutive timeout threshold is 3.

The configuration is saved on the device rather than the mobile communication modem.

Configuring manual reboot

About this task

A mobile communication modem can automatically detect running errors and reboot. If the mobile communication modem fails to reboot by itself, you can use this command to manually reboot it.

Procedure

1. Enter system view.
   `system-view`
2. Enter cellular interface view.
   `controller cellular` *`interface-number`*
3. Manually reboot the mobile communication modem.
   `modem reboot`

△ **CAUTION:**

Execute this command with caution. This command will disconnect the established 3G/4G modem connections.

Display and maintenance commands for mobile communication modem management

Execute **`display`** commands in any view and the **`reset`** command in user view.

Task	Command
Display the call connection information for the mobile communication modem.	**display cellular** [*interface-number*]
Display information about a cellular interface.	**display controller** [**cellular** [*interface-number*]]
Display information about an Eth-channel interface.	**display interface** [**eth-channel** [*channel-id*]] [**brief** [**description** | **down**]]
Clear the statistics for a cellular interface.	**reset counters controller** [**cellular** [*interface-number*]]
Clear the statistics for an Eth-channel interface.	**reset counters interface** [**eth-channel** [*channel-id*]]

Troubleshooting

Symptom

A mobile communication modem fails to function correctly. For example, the mobile communication modem receives no signals or fails to connect to service providers' networks.

Solution

To resolve the issue:

1. Execute the **shutdown** command and the **undo shutdown** command on the cellular interface.
2. If the mobile communication modem still fails to function, execute the **modem reboot** command on the cellular interface.
3. If the issue persists, contact NSFOCUS Support.

NSFOCUS Firewall Series

NF Layer 3—IP Services

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for IP services features, including ARP, IP addressing, DHCP, DNS, IP forwarding basics, fast forwarding, adjacency table, IP performance optimization, IPv6 basics, DHCPv6, IPv6 fast forwarding, and multi-CPU packet distribution.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

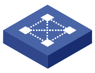
GUI conventions

Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring IP addressing

About IP addressing

The IP addresses in this chapter refer to IPv4 addresses unless otherwise specified.

IP address representation and classes

IP addressing uses a 32-bit address to identify each host on an IPv4 network. To make addresses easier to read, they are written in dotted decimal notation, each address being four octets in length. For example, address 00001010000000010000000100000001 in binary is written as 10.1.1.1.

Each IP address breaks down into the following sections:

- **Net ID**—Identifies a network. The first several bits of a net ID, known as the class field or class bits, identify the class of the IP address.
- **Host ID**—Identifies a host on a network.

IP addresses are divided into five classes, as shown in Figure 1. The shaded areas represent the address class. The first three classes are most commonly used.

Figure 1 IP address classes

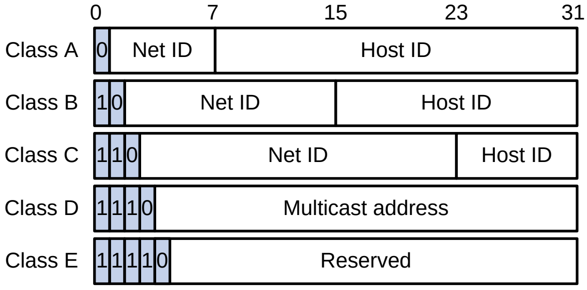


Table 1 IP address classes and ranges

Class	Address range	Remarks
A	0.0.0.0 to 127.255.255.255	The IP address 0.0.0.0 is used by a host at startup for temporary communication. This address is never a valid destination address. Addresses starting with 127 are reserved for loopback test. Packets destined to these addresses are processed locally as input packets rather than sent to the link.
B	128.0.0.0 to 191.255.255.255	N/A
C	192.0.0.0 to 223.255.255.255	N/A
D	224.0.0.0 to 239.255.255.255	Multicast addresses.
E	240.0.0.0 to 255.255.255.255	Reserved for future use, except for the broadcast address 255.255.255.255.

Special IP addresses

The following IP addresses are for special use and cannot be used as host IP addresses:

- **IP address with an all-zero net ID**—Identifies a host on the local network. For example, IP address 0.0.0.16 indicates the host with a host ID of 16 on the local network.
- **IP address with an all-zero host ID**—Identifies a network.
- **IP address with an all-one host ID**—Identifies a directed broadcast address. For example, a packet with the destination address of 192.168.1.255 will be broadcast to all the hosts on the network 192.168.1.0.

Subnetting and masking

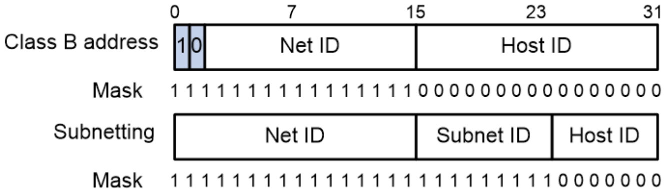
Subnetting divides a network into smaller networks called subnets by using some bits of the host ID to create a subnet ID.

Masking identifies the boundary between the host ID and the combination of net ID and subnet ID.

Each subnet mask comprises 32 bits that correspond to the bits in an IP address. In a subnet mask, consecutive ones represent the net ID and subnet ID, and consecutive zeros represent the host ID.

Before being subnetted, Class A, B, and C networks use these default masks (also called natural masks): 255.0.0.0, 255.255.0.0, and 255.255.255.0, respectively.

Figure 2 Subnetting a Class B network



Subnetting increases the number of addresses that cannot be assigned to hosts. Therefore, using subnets means accommodating fewer hosts.

For example, a Class B network without subnetting can accommodate 1022 more hosts than the same network subnetted into 512 subnets.

- **Without subnetting**—65534 (2^{16} – 2) hosts. (The two deducted addresses are the broadcast address, which has an all-one host ID, and the network address, which has an all-zero host ID.)
- **With subnetting**—Using the first nine bits of the host-id for subnetting provides 512 (2^9) subnets. However, only seven bits remain available for the host ID. This allows 126 (2^7 – 2) hosts in each subnet, a total of 64512 (512 × 126) hosts.

IP address assignment

The following are methods available for assigning an IP address to an interface:

- Manual assignment. This chapter describes only manual IP address assignment for interfaces.
- BOOTP. For information about BOOTP, see "Configuring the BOOTP client."
- DHCP. For information about DHCP, see "Configuring the DHCP client."
- PPP address negotiation. For information about PPP, see *Layer 2—WAN Access Configuration Guide*.

These methods are mutually exclusive. If you change the IP address assignment method, the new IP address will overwrite the previous address.

Assigning an IP address to an interface

About this task

An interface can have one primary address and multiple secondary addresses.

Typically, you need to configure a primary IP address for an interface. If the interface connects to multiple subnets, configure primary and secondary IP addresses on the interface so the subnets can communicate with each other through the interface.

Restrictions and guidelines

- An interface can have only one primary IP address. If you execute the **`ip address`** command multiple times to specify different primary IP addresses, the most recent configuration takes effect.
- You cannot assign secondary IP addresses to an interface that obtains an IP address through BOOTP, DHCP, PPP address negotiation, or IP unnumbered.
- The primary and secondary IP addresses assigned to the interface can be located on the same network segment. Different interfaces on your device must reside on different network segments.
- You can assign interfaces IP addresses that have different masks but the same network address if ANDed with the shortest mask. For example, 1.1.1.1/16 and 1.1.2.1/24 have the same network address 1.1.0.0 if ANDed with 255.255.0.0. You can assign the IP addresses to two interfaces on the device. By default, users connected to the two interfaces cannot communicate with each other. For the users to communicate, you must configure common proxy ARP on the device. For more information, see "Configuring proxy ARP."

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Assign an IP address to the interface.

   `ip address` *`ip-address`* { *`mask-length`* | *`mask`* } [**`sub`**]

   By default, no IP address is assigned to the interface.

Configuring IP unnumbered

About this task

You can configure an interface to borrow an IP address from other interfaces. This is called IP unnumbered, and the interface borrowing the IP address is called IP unnumbered interface.

You can use IP unnumbered to save IP addresses when available IP addresses are inadequate or when an interface is used only occasionally.

Restrictions and guidelines

- Loopback interfaces cannot borrow IP addresses of other interfaces, but other interfaces can borrow IP addresses of loopback interfaces.
- An interface cannot borrow an IP address from an unnumbered interface.
- Multiple interfaces can use the same unnumbered IP address.
- If an interface has multiple manually configured IP addresses, only the manually configured primary IP address can be borrowed.

- A dynamic routing protocol cannot be enabled on the interface where IP unnumbered is configured. To enable the interface to communicate with other devices, configure a static route to the peer device on the interface.
- A Layer 3 Ethernet subinterface or Layer 3 aggregate subinterface cannot learn the ARP entry for the IP address it borrows. As a best practice, do not configure IP unnumbered on Layer 3 Ethernet subinterfaces or Layer 3 aggregate subinterfaces.

Prerequisites

Assign an IP address to the interface from which you want to borrow the IP address. Alternatively, you can configure the interface to obtain one through BOOTP, DHCP, or PPP address negotiation. .

Do not configure IP unnumbered on an Ethernet subinterface or a Layer-3 aggregate subinterface because they cannot learn the ARP entries for the borrowed IP address.

Procedure

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Specify the interface to borrow the IP address of the specified interface.
   ip address unnumbered interface *interface-type interface-number*
   By default, the interface does not borrow IP addresses from other interfaces.

Display and maintenance commands for IP addressing

Execute **display** commands in any view.

Task	Command
Display brief IP configuration for Layer 3 interfaces.	**display ip interface** [*interface-type* [*interface-number*]] **brief** [**description**]
Display IP configuration and statistics for the specified or all Layer 3 interfaces.	**display ip interface** [*interface-type* [*interface-number*]]

Contents

Configuring IP forwarding basic settings

About FIB table

A device uses the FIB table to make packet forwarding decisions.

A device selects optimal routes from the routing table, and puts them into the FIB table. Each FIB entry specifies the next hop IP address and output interface for packets destined for a specific subnet or host.

For more information about the routing table, see *Layer 3—IP Routing Configuration Guide*.

Use the **display fib** command to display the FIB table. The following example displays the entire FIB table.

```
<Sysname> display fib

Destination count: 4 FIB entry count: 4

Flag:
  U:Useable   G:Gateway   H:Host   B:Blackhole   D:Dynamic   S:Static
  R:Relay     F:FRR

Destination/Mask   Nexthop         Flag     OutInterface/Token       Label
10.2.0.0/16        10.2.1.1        U        GE1/0/1                  Null
10.2.1.1/32        127.0.0.1       UH       InLoop0                  Null
127.0.0.0/8        127.0.0.1       U        InLoop0                  Null
127.0.0.1/32       127.0.0.1       UH       InLoop0                  Null
```

A FIB entry includes the following items:

- **Destination**—Destination IP address.
- **Mask**—Network mask. The mask and the destination address identify the destination network. A logical AND operation between the destination address and the network mask yields the address of the destination network. For example, if the destination address is 192.168.1.40 and the mask 255.255.255.0, the address of the destination network is 192.168.1.0. A network mask includes a certain number of consecutive 1s. It can be expressed in dotted decimal format or by the number of the 1s.
- **Nexthop**—IP address of the next hop.
- **Flag**—Route flag.
- **OutInterface**—Output interface.
- **Token**—MPLS Label Switched Path index number.
- **Label**—Inner label.

Enabling last hop holding

About this task

Last hop holding implements symmetric routing.

When the interface enabled with this feature receives the first IP packet of a forward flow, this feature implements the following operations:

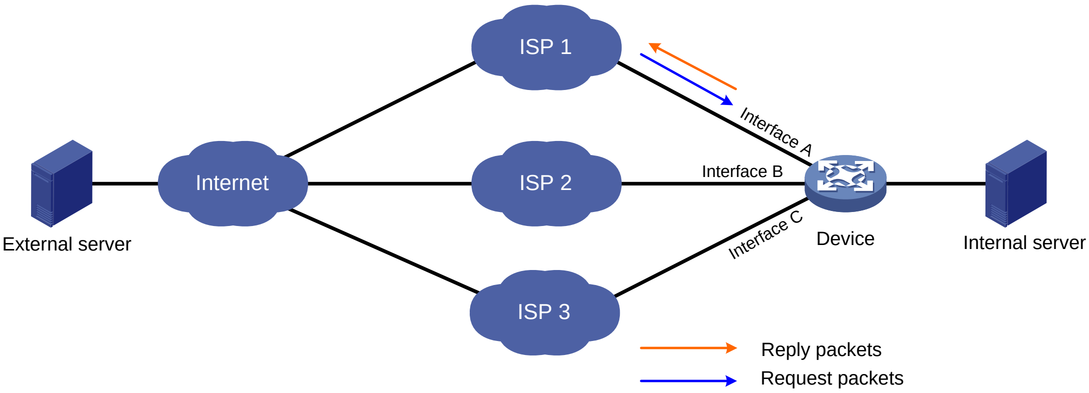
- Obtains the forward flow information and last hop information of the packet.

- Based on the obtained information, creates a fast forwarding entry for the return flow.

When packets of the return flow arrive at the device, the device forwards those packets according to the entry.

As is shown in Figure 1, when the external server sends a request to the internal server, the packet travels through ISP 1 to Interface A on the device. The last hop holding feature on the device ensures that the reply packet follows the same route as the request packet back to ISP 1. If last hop holding is disabled, the reply packet might be sent out of Interface B or Interface C to the external network.

Figure 1 Last hop holding application



Restrictions and guidelines

This feature relies on fast forwarding entries. If the MAC address of a last hop changes on an Ethernet link, this feature can function correctly only after the fast forwarding entry is updated for the MAC address.

In an IRF fabric, this feature is not supported for those packets that are forwarded between member devices.

This feature is not applicable to asymmetric-path traffic forwarding in an RBM-based hot backup system. For more information about RBM-based hot backup, see high availability group configuration in *High Availability Configuration Guide*.

On a device that supports deployment of multiple security service modules, this feature does not take effect on the external traffic destined for that device

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type`* { *`interface-number`* | *`interface-number.subnumber`* }

3. Enable last hop holding.

   `ip last-hop hold`

   By default, last hop holding is disabled.

Enabling last hop backup

About this task

This feature enables the system to transmit the forward flow and reverse flow between the local node and a peer node over the same path.

In an IRF fabric enabled with this feature, the IRF master device performs the following operations when receiving the first IP packet of a forward flow on an interface enabled with last hop holding:

1. Saves the last hop information of the packet.
2. Synchronizes the last hop information to subordinate devices in the IRF fabric.

The last hop information can be used for guiding the backward flow when the flow arrives at the master device or is forwarded through a subordinate device.

For this feature to take effect in an IRF fabric, you must also enable session synchronization by using the **session synchronization enable** command. For more information about the **session synchronization enable** command, see *Security Command Reference*.

This feature is also applicable to multi-module devices enabled with service backup. If this feature is enabled on such a device, a device module performs the following operations when receiving the first IP packet of a forward flow on an interface enabled with last hop holding:

1. Saves the last hop information of the packet.
2. Synchronizes the last hop information to other modules in the device.

The last hop information can be used for guiding the backward flow when the flow arrives at one of these modules.

For this feature to take effect on a multi-module device, you must also enable session flow redirection by using the **session flow-redirect enable** command. For more information about the **session flow-redirect enable** command, see *Security Command Reference*.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enable last hop backup.
   ```
   last-hop backup enable
   ```
   By default, last hop backup is enabled.

Display and maintenance commands for FIB table

Execute **display** commands in any view.

Task	Command
Display FIB entries.	**display fib** [**vpn-instance** *vpn-instance-name*] [*ip-address* [*mask* | *mask-length*]]

Configuring load sharing

About load sharing

If a routing protocol finds multiple equal-cost best routes to the same destination, the device forwards packets over the equal-cost routes to implement load sharing.

Configuring load sharing mode

About this task

In the per-flow load sharing mode, the device forwards flows over equal-cost routes. Packets of one flow travel along the same routes. You can configure the device to identify a flow based on the following criteria: source IP address, destination IP address, source port number, destination port number, and IP protocol number.

In the per-packet load sharing mode, the device forwards packets over equal-cost routes.

Hardware and feature compatibility

Models	Command compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB1180, NFNX3-HDB1480	No

Restrictions and guidelines

Per-packet load sharing does not take effect on packets that are fast forwarded. It takes effect only on those packets that are delivered to the CPU. For more information about configuring load balancing for fast forwarding, see fast forwarding configuration in *Layer 3—IP Services Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Configure load sharing.

   ip load-sharing mode { **per-flow** [**dest-ip** | **dest-port** | **ip-pro** | **src-ip** | **src-port**] * | **per-packet** } { **global** | **slot** *slot-number* }

   By default, the device performs per-flow load sharing.

Enabling local-first load sharing

About this task

Local-first load sharing distributes traffic preferentially across the output interfaces on the receiving IRF member device if output interfaces for multiple equal-cost routes are on different members. This feature enhances packets forwarding efficiency.

Procedure

1. Enter system view.

```
system-view
```

2. Enable local-first load sharing.

```
ip load-sharing local-first enable
```

By default, local-first load sharing is disabled.

Contents

Configuring fast forwarding

About fast forwarding

Fast forwarding reduces route lookup time and improves packet forwarding efficiency by using a high-speed cache and data-flow-based technology. It identifies a data flow by using the following fields: source IP address, source port number, destination IP address, destination port number, and protocol number. After a flow's first packet is forwarded through the routing table, fast forwarding creates an entry and uses the entry to forward subsequent packets of the flow.

Restrictions and guidelines: Fast forwarding configuration

Fast forwarding can process fragmented IP packets, but it does not fragment IP packets.

Fast forwarding can be implemented by software or hardware. Unless otherwise noted, fast forwarding in this chapter refers to software fast forwarding.

Configuring the aging time for fast forwarding entries

About this task

The fast forwarding table uses an aging timer for each forwarding entry. If an entry is not updated before the timer expires, the device deletes the entry. If an entry has a hit within the aging time, the aging timer restarts.

Procedure

1. Enter system view.

   `system-view`

2. Configure the aging time for fast forwarding entries.

   `ip fast-forwarding aging-time` *`aging-time`*

   By default, the aging time is 30 seconds.

Configuring fast forwarding load sharing

About this task

Fast forwarding load sharing enables the device to identify a data flow by using the packet information.

If fast forwarding load sharing is disabled, the device identifies a data flow by the packet information and the input interface.

Procedure

1. Enter system view.

   `system-view`

2. Configure fast forwarding load sharing. Choose one option as needed:
   - Enable fast forwarding load sharing.
     ip fast-forwarding load-sharing
   - Disable fast forwarding load sharing.
     undo ip fast-forwarding load-sharing

   By default, fast forwarding load sharing is enabled.

Enabling DSCP-based fast forwarding for GRE packets

About this task

This feature uses the DSCP value in the outer header instead of the source port number among the identification criteria to identify GRE traffic flows.

Procedure

1. Enter system view.
   system-view
2. Enable DSCP-based fast forwarding for GRE packets.
   ip fast-forwarding dscp

   By default, DSCP-based fast forwarding for GRE packet is disabled.

Display and maintenance commands for fast forwarding

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the aging time of fast forwarding entries.	**display ip fast-forwarding aging-time**
Display fast forwarding entries.	**display ip fast-forwarding cache** [*ip-address*] [**slot** *slot-number*]
Display fast forwarding entries about fragmented packets.	**display ip fast-forwarding fragcache** [*ip-address*] [**slot** *slot-number*]
Clear the fast forwarding table.	**reset ip fast-forwarding cache** [**slot** *slot-number*]

Contents

Configuring ARP

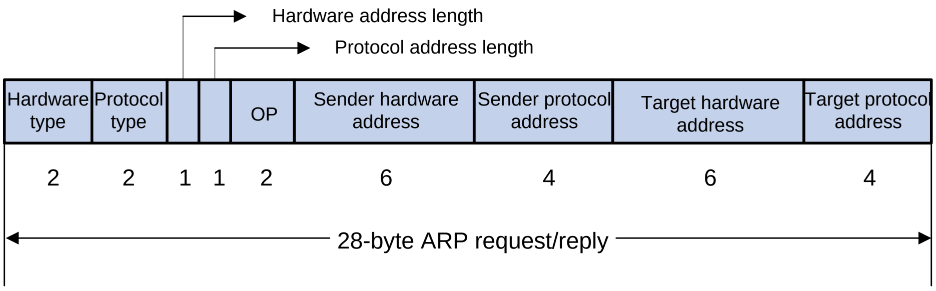
About ARP

ARP resolves IP addresses into MAC addresses on Ethernet networks.

ARP message format

ARP uses two types of messages: ARP request and ARP reply. Figure 1 shows the format of ARP request/reply messages. Numbers in the figure refer to field lengths.

Figure 1 ARP message format



- **Hardware type**—Hardware address type. The value 1 represents Ethernet.
- **Protocol type**—Type of the protocol address to be mapped. The hexadecimal value 0x0800 represents IP.
- **Hardware address length and protocol address length**—Length, in bytes, of a hardware address and a protocol address. For an Ethernet address, the value of the hardware address length field is 6. For an IPv4 address, the value of the protocol address length field is 4.
- **OP**—Operation code, which describes the type of ARP message. The value 1 represents an ARP request, and the value 2 represents an ARP reply.
- **Sender hardware address**—Hardware address of the device sending the message.
- **Sender protocol address**—Protocol address of the device sending the message.
- **Target hardware address**—Hardware address of the device to which the message is being sent.
- **Target protocol address**—Protocol address of the device to which the message is being sent.

ARP operating mechanism

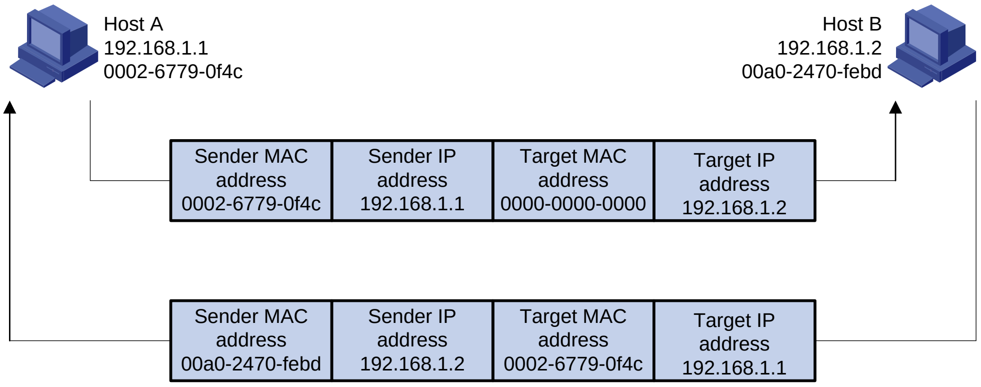
As shown in Figure 2, Host A and Host B are on the same subnet. Host A sends a packet to Host B as follows:

1. Host A looks through the ARP table for an ARP entry for Host B. If one entry is found, Host A uses the MAC address in the entry to encapsulate the IP packet into a data link layer frame. Then Host A sends the frame to Host B.
2. If Host A finds no entry for Host B, Host A buffers the packet and broadcasts an ARP request. The payload of the ARP request contains the following information:
   - **Sender IP address and sender MAC address**—Host A's IP address and MAC address.
   - **Target IP address**—Host B's IP address.
   - **Target MAC address**—An all-zero MAC address.

All hosts on this subnet can receive the broadcast request, but only the requested host (Host B) processes the request.

3. Host B compares its own IP address with the target IP address in the ARP request. If they are the same, Host B operates as follows:
   a. Adds the sender IP address and sender MAC address into its ARP table.
   b. Encapsulates its MAC address into an ARP reply.
   c. Unicasts the ARP reply to Host A.
4. After receiving the ARP reply, Host A operates as follows:
   a. Adds the MAC address of Host B into its ARP table.
   b. Encapsulates the MAC address into the packet and sends the packet to Host B.

Figure 2 ARP address resolution process



Sender MAC address	Sender IP address	Target MAC address	Target IP address
0002-6779-0f4c	192.168.1.1	0000-0000-0000	192.168.1.2

Sender MAC address	Sender IP address	Target MAC address	Target IP address
00a0-2470-febd	192.168.1.2	0002-6779-0f4c	192.168.1.1

If Host A and Host B are on different subnets, Host A sends a packet to Host B as follows:

1. Host A broadcasts an ARP request where the target IP address is the IP address of the gateway.
2. The gateway responds with its MAC address in an ARP reply to Host A.
3. Host A uses the gateway's MAC address to encapsulate the packet, and then sends the packet to the gateway.
4. If the gateway has an ARP entry for Host B, it forwards the packet to Host B directly. If not, the gateway broadcasts an ARP request, in which the target IP address is the IP address of Host B.
5. After the gateway gets the MAC address of Host B, it sends the packet to Host B.

ARP entry types

An ARP table stores dynamic ARP entries Rule ARP entries, and static ARP entries.

Dynamic ARP entry

ARP automatically creates and updates dynamic entries. A dynamic ARP entry is removed when its aging timer expires or the output interface goes down. In addition, a dynamic ARP entry can be overwritten by a static ARP entry.

Static ARP entry

A static ARP entry is manually configured and maintained. It does not age out and cannot be overwritten by any dynamic ARP entry.

Static ARP entries protect communication between devices because attack packets cannot modify the IP-to-MAC mapping in a static ARP entry.

The device supports the following types of static ARP entries:

- **Long static ARP entry**—It is directly used for forwarding packets. A long static ARP entry contains the IP address, MAC address, and one of the following combinations:
  - VLAN and output interface.
  - Input and output interfaces.
- **Short static ARP entry**—It contains only the IP address and MAC address.

  If the output interface is a Layer 3 Ethernet interface, the short ARP entry can be directly used to forward packets.

  If the output interface is a VLAN interface, the device sends an ARP request whose target IP address is the IP address in the short entry. If the sender IP and MAC addresses in the received ARP reply match the short static ARP entry, the device performs the following operations:
  - Adds the interface that received the ARP reply to the short static ARP entry.
  - Uses the resolved short static ARP entry to forward IP packets.

To communicate with a host by using a fixed IP-to-MAC mapping, configure a short static ARP entry on the device. To communicate with a host by using a fixed IP-to-MAC mapping through an interface in a VLAN, configure a long static ARP entry on the device.

Rule ARP entry

Rule ARP entries can be directly used for packet forwarding. A Rule ARP entry does not age out, and it cannot be updated. It can be overwritten by a static ARP entry.

ARP creates Rule ARP entries by learning from the Portal module. For more information about portal, see *Security Configuration Guide*.

- IPoE. For more information about IPoE, see *Security Configuration Guide*.
- Portal. For more information about portal, see *Security Configuration Guide*.

ARP tasks at a glance

All ARP tasks are optional.

- Configuring a static ARP entry
  - Configuring a short static ARP entry
  - Configuring a long static ARP entry
- Configuring features for dynamic ARP entries
  - Setting the dynamic ARP learning limit for a device
  - Setting the dynamic ARP learning limit for an interface
  - Setting the aging timer for dynamic ARP entries
  - Enabling dynamic ARP entry check
- Enabling an IP unnumbered interface to learn ARP entries for different subnets
- Enabling ARP logging

Configuring a static ARP entry

Static ARP entries are effective when the device functions correctly.

Configuring a short static ARP entry

Restrictions and guidelines

A resolved short static ARP entry becomes unresolved upon certain events, for example, when the resolved output interface goes down, or the corresponding VLAN or VLAN interface is deleted.

Procedure

1. Enter system view.

   system-view

2. Configure a short static ARP entry.

   arp static *ip-address mac-address* [**vpn-instance** *vpn-instance-name*] [**description** *text*]

Configuring a long static ARP entry

About this task

Long static ARP entries can be effective or ineffective. Ineffective long static ARP entries cannot be used for packet forwarding. A long static ARP entry is ineffective when any of the following conditions exists:

- The IP address in the entry conflicts with a local IP address.
- No local interface has an IP address in the same subnet as the IP address in the ARP entry.

A long static ARP entry in a VLAN is deleted if the VLAN or VLAN interface is deleted.

Procedure

1. Enter system view.

   system-view

2. Configure a long static ARP entry.

   arp static *ip-address mac-address* [*vlan-id interface-type interface-number*] [**vpn-instance** *vpn-instance-name*] [**description** *text*]

Configuring features for dynamic ARP entries

Setting the dynamic ARP learning limit for a device

About this task

A device can dynamically learn ARP entries. To prevent a device from holding too many ARP entries, you can set the maximum number of dynamic ARP entries that the device can learn. When the limit is reached, the device stops ARP learning.

If you set a value lower than the number of existing dynamic ARP entries, the device does not delete the existing entries unless they age out. You can use the **reset arp dynamic** command to clear dynamic ARP entries.

Procedure

1. Enter system view.

   system-view

2. Set the dynamic ARP learning limit for the device.

   arp max-learning-number *max-number* **slot** *slot-number*

   By default, the device can learn a maximum of 16384 dynamic ARP entries.

   To disable the device from dynamic ARP learning, set the value to 0.

Setting the dynamic ARP learning limit for an interface

About this task

An interface can dynamically learn ARP entries. To prevent an interface from holding too many ARP entries, you can set the maximum number of dynamic ARP entries that the interface can learn. When the limit is reached, the interface stops ARP learning.

You can set limits for both a Layer 2 interface and the VLAN interface for a permitted VLAN on the Layer 2 interface. The Layer 2 interface learns an ARP entry only when neither limit is reached.

The total dynamic ARP learning limit for all interfaces will not be higher than the dynamic ARP learning limit for the device.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Set the dynamic ARP learning limit for the interface.

   `arp max-learning-num` *`max-number`*

   By default, an interface can learn a maximum of 16384 dynamic ARP entries.

   To disable the interface from dynamic ARP learning, set the value to 0.

Setting the aging timer for dynamic ARP entries

About this task

Each dynamic ARP entry in the ARP table has a limited lifetime, called an aging timer. The aging timer of a dynamic ARP entry is reset each time the dynamic ARP entry is updated. A dynamic ARP entry that is not updated before its aging timer expires is deleted from the ARP table.

Procedure

1. Enter system view.

   `system-view`

2. Set the aging timer for dynamic ARP entries.

   `arp timer aging` *`aging-time`*

   The default setting is 20 minutes.

Enabling dynamic ARP entry check

About this task

The dynamic ARP entry check feature disables the device from supporting dynamic ARP entries that contain multicast MAC addresses. The device cannot learn dynamic ARP entries containing multicast MAC addresses. You cannot manually add static ARP entries containing multicast MAC addresses.

When dynamic ARP entry check is disabled, ARP entries containing multicast MAC addresses are supported. The device can learn dynamic ARP entries containing multicast MAC addresses obtained from the ARP packets sourced from a unicast MAC address. You can also manually add static ARP entries containing multicast MAC addresses.

Procedure

1. Enter system view.

   `system-view`

2. Enable dynamic ARP entry check.

   `arp check enable`

   By default, dynamic ARP entry check is enabled.

Enabling an IP unnumbered interface to learn ARP entries for different subnets

About this task

An IP unnumbered interface cannot learn the ARP entry of the peer device if the unnumbered interface and the peer device are on different subnets. To ensure communication between them, you can enable this feature on the IP unnumbered interface.

If an IP unnumbered interface is disabled from learning ARP entries for different subnets, existing ARP entries learned for different subnets are deleted after they age out.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the interface to borrow the IP address of the specified interface.

   `ip address unnumbered interface` *`interface-type interface-number`*

   By default, the interface does not borrow IP addresses from other interfaces.

4. Enable the IP unnumbered interface to learn ARP entries for different subnets.

   `arp ip-unnumbered learning enable`

   By default, an IP unnumbered interface cannot learn ARP entries for different subnets.

Enabling ARP logging

About this task

This feature enables a device to log ARP events when ARP cannot resolve IP addresses correctly. The log information helps administrators locate and solve problems. The device can log the following ARP events:

- On a proxy ARP-disabled interface, the target IP address of a received ARP packet is not one of the following IP addresses:
  - The IP address of the receiving interface.
  - The virtual IP address of the VRRP group.
  - The public IP address after NAT.
- The sender IP address of a received ARP reply conflicts with one of the following IP addresses:
  - The IP address of the receiving interface.
  - The virtual IP address of the VRRP group.
  - The public IP address after NAT.

The device sends ARP log messages to the information center. You can use the **info-center source** command to specify the log output rules for the information center. For more information about information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable ARP logging.

   arp check log enable

   By default, ARP logging is disabled.

Display and maintenance commands for ARP

△ **CAUTION:**

Clearing ARP entries from the ARP table might cause communication failures. Make sure the entries to be cleared do not affect current communications.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display ARP entries.	**display arp** [[**all** | **dynamic** | **static**] [**slot** *slot-number*] | **vlan** *vlan-id* | **interface** *interface-type interface-number*] [**count** | **verbose**]
Display the ARP entry for an IP address.	**display arp** *ip-address* [**slot** *slot-number*] [**verbose**]
Display the aging timer of dynamic ARP entries.	**display arp timer aging**
Display the ARP entries for a VPN instance.	**display arp vpn-instance** *vpn-instance-name* [**count** | **verbose**]
Clear ARP entries from the ARP table.	**reset arp** { **all** | **dynamic** | **interface** *interface-type interface-number* | **slot** *slot-number* | **static** }

Configuring gratuitous ARP

About gratuitous ARP

In a gratuitous ARP packet, the sender IP address and the target IP address are the IP address of the sending device.

A device sends a gratuitous ARP packet for either of the following purposes:

- Determine whether its IP address is already used by another device. If the IP address is already used, the device is informed of the conflict by an ARP reply.
- Inform other devices of a MAC address change.

IP conflict detection

When an interface obtains an IP address, the device broadcasts gratuitous ARP packets in the LAN where the interface resides. If the device receives an ARP reply, its IP address conflicts with the IP address of another device in the LAN. The device displays a log message about the conflict and informs the administrator to change the IP address. The device will not use the conflicting IP address. If no ARP reply is received, the device uses the IP address.

Gratuitous ARP packet learning

This feature enables a device to create or update ARP entries by using the sender IP and MAC addresses in received gratuitous ARP packets.

When this feature is disabled, the device uses received gratuitous ARP packets to update existing ARP entries only. ARP entries are not created based on the received gratuitous ARP packets, which saves ARP table space.

Periodic sending of gratuitous ARP packets

Periodic sending of gratuitous ARP packets helps downstream devices update ARP entries or MAC entries in a timely manner.

This feature can implement the following functions:

- Prevent gateway spoofing.

  Gateway spoofing occurs when an attacker uses the gateway address to send gratuitous ARP packets to the hosts on a network. The traffic destined for the gateway from the hosts is sent to the attacker instead. As a result, the hosts cannot access the external network.

  To prevent such gateway spoofing attacks, you can enable the gateway to send gratuitous ARP packets at intervals. Gratuitous ARP packets contain the primary IP address and manually configured secondary IP addresses of the gateway, so hosts can learn correct gateway information.

- Prevent ARP entries from aging out.

  If network traffic is heavy or if the host CPU usage is high, received ARP packets can be discarded or are not promptly processed. Eventually, the dynamic ARP entries on the receiving host age out. The traffic between the host and the corresponding devices is interrupted until the host re-creates the ARP entries.

  To prevent this problem, you can enable the gateway to send gratuitous ARP packets periodically. Gratuitous ARP packets contain the primary IP address and manually configured secondary IP addresses of the gateway, so the receiving hosts can update ARP entries in a timely manner.

- Prevent the virtual IP address of a VRRP group from being used by a host.

  The master router of a VRRP group can periodically send gratuitous ARP packets to the hosts on the local network. The hosts can then update local ARP entries and avoid using the virtual IP address of the VRRP group. The sender MAC address in the gratuitous ARP packet is the virtual MAC address of the virtual router.

Gratuitous ARP tasks at a glance

All gratuitous ARP tasks are optional. If all of the following features are disabled, gratuitous ARP still provides the IP conflict detection function.

- Enabling IP conflict notification
- Enabling gratuitous ARP packet learning
- Enabling periodic sending of gratuitous ARP packets
- Enabling sending gratuitous ARP packets for ARP requests with sender IP address on a different subnet
- Configuring gratuitous ARP packet retransmission for the device MAC address change

Enabling IP conflict notification

About this task

Upon detecting an IP conflict, the device will sends a gratuitous ARP request. By default, the device displays an error message only after it receives an ARP reply. You can enable this feature to allow the device to display an error message immediately upon detecting an IP conflict.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable IP conflict notification.

   ```
   arp ip-conflict log prompt
   ```

   If the system starts up with the factory defaults, IP conflict notification is enabled. If the system starts up with the initial configuration, IP conflict notification is disabled. For more information about factory defaults and initial configuration, see configuration file management in *Fundamentals Configuration Guide*.

Enabling gratuitous ARP packet learning

1. Enter system view.

   ```
   system-view
   ```

2. Enable gratuitous ARP packet learning.

   ```
   gratuitous-arp-learning enable
   ```

   By default, gratuitous ARP packet learning is enabled.

Enabling periodic sending of gratuitous ARP packets

Restrictions and guidelines

- You can enable periodic sending of gratuitous ARP packets on a maximum of 1024 interfaces.
- Periodic sending of gratuitous ARP packets takes effect on an interface only when the following conditions are met:
  - The data link layer state of the interface is up.
  - The interface has an IP address.
- If you change the sending interval for gratuitous ARP packets, the configuration takes effect at the next sending interval.
- The sending interval for gratuitous ARP packets might be much longer than the specified sending interval in any of the following circumstances:
  - This feature is enabled on multiple interfaces.
  - Each interface is configured with multiple secondary IP addresses.
  - A small sending interval is configured when the previous two conditions exist.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable periodic sending of gratuitous ARP packets.

   arp send-gratuitous-arp [**interval** *interval*]

   By default, periodic sending of gratuitous ARP packets is disabled.

Enabling sending gratuitous ARP packets for ARP requests with sender IP address on a different subnet

1. Enter system view.

   system-view

2. Enable the device to send gratuitous ARP packets upon receiving ARP requests whose sender IP address belongs to a different subnet.

   gratuitous-arp-sending enable

   By default, a device does not send gratuitous ARP packets upon receiving ARP requests whose sender IP address belongs to a different subnet.

Configuring gratuitous ARP packet retransmission for the device MAC address change

About this task

The device sends a gratuitous ARP packet to inform other devices of its MAC address change. However, the other devices might fail to receive the packet because the device sends the gratuitous ARP packet once only by default. Configure the gratuitous ARP packet retransmission feature to ensure that the other devices can receive the packet.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the times and the interval for retransmitting a gratuitous ARP packet for the device MAC address change.

   ```
   gratuitous-arp mac-change retransmit times interval seconds
   ```

   By default, the device sends a gratuitous packet to inform its MAC address change once only.

Configuring proxy ARP

About proxy ARP

Proxy ARP enables a device on one network to answer ARP requests for an IP address on another network. With proxy ARP, hosts on different broadcast domains can communicate with each other as they would on the same broadcast domain.

Proxy ARP includes common proxy ARP and local proxy ARP.

- **Common proxy ARP**—Allows communication between hosts that connect to different Layer 3 interfaces and reside in different broadcast domains.
- **Local proxy ARP**—Allows communication between hosts that connect to the same Layer 3 interface and reside in different broadcast domains.

Enabling common proxy ARP

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   The following interface types are supported:

   - VLAN interface.
   - Layer 3 Ethernet interface.
   - Layer 3 Ethernet subinterface.
   - Layer 3 aggregate interface.
   - Layer 3 aggregate subinterface.

3. Enable common proxy ARP.

   `proxy-arp enable`

   By default, common proxy ARP is disabled.

Enabling local proxy ARP

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   The following interface types are supported:

   - VLAN interface.
   - Layer 3 Ethernet interface.
   - Layer 3 Ethernet subinterface.
   - Layer 3 aggregate interface.
   - Layer 3 aggregate subinterface.

3. Enable local proxy ARP.

   `local-proxy-arp enable` [**`ip-range`** *`start-ip-address`* **`to`** *`end-ip-address`*]

By default, local proxy ARP is disabled.

Display and maintenance commands for proxy ARP

Execute **display** commands in any view.

Task	Command
Display common proxy ARP status.	**display proxy-arp** [**interface** *interface-type interface-number*]
Display local proxy ARP status.	**display local-proxy-arp** [**interface** *interface-type interface-number*]

Configuring ARP snooping

About ARP snooping

ARP snooping is used in Layer 2 switching networks. It creates ARP snooping entries by using information in ARP packets.

Creation of ARP snooping entries

If you enable ARP snooping for a VLAN, ARP packets received in the VLAN are redirected to the CPU. The CPU uses the sender IP and MAC addresses of the ARP packets, and receiving VLAN and port to create ARP snooping entries.

Aging of ARP snooping entries

The aging timer and valid period of an ARP snooping entry are 25 minutes and 15 minutes. If an ARP snooping entry is not updated in 12 minutes, the device sends an ARP request. The ARP request uses the IP address of the entry as the target IP address. If an ARP snooping entry is not updated in 15 minutes, it becomes invalid and cannot be used. After that, if an ARP packet matching the entry is received, the entry becomes valid, and its aging timer restarts.

If the aging timer of an ARP snooping entry expires, the entry is removed.

Protection for ARP snooping

An attack occurs if an ARP packet has the same sender IP address as a valid ARP snooping entry but a different sender MAC address. The ARP snooping entry becomes invalid, and it is removed in 1 minute.

Enabling ARP snooping

1. Enter system view.
   `system-view`
2. Enter VLAN view.
   `vlan` *`vlan-id`*
3. Enable ARP snooping
   `arp snooping enable`
   By default, ARP snooping is disabled.

Display and maintenance commands for ARP snooping

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display ARP snooping entries.	**display arp snooping** [**vlan** *vlan-id*] [**slot** *slot-number*] [**count**] **display arp snooping ip** *ip-address* [**slot** *slot-number*]
Delete ARP snooping entries.	**reset arp snooping** [**ip** *ip-address* | **vlan** *vlan-id*]

Configuring ARP fast-reply

About ARP fast-reply

ARP fast-reply enables a device to directly answer ARP requests according to IPSG bindings or ARP snooping entries. ARP fast-reply functions in a VLAN.

The device processes a received ARP request as follows:

1. Checks the target IP address of the ARP request.
   - If the target IP address is the IP address of the VLAN interface, the device delivers the request to the ARP module.
   - If the target IP address is not the IP address of the VLAN interface, the process goes to step 2.
2. Searches IPSG bindings by using the target IP address.
   - If a match is found and the interface in the matching entry is a wireless interface, the device returns a reply according to the matching entry.
   - If a match is found and the interface in the matching entry is the Ethernet interface that received the ARP request, the device does not return any reply.
   - If a match is found and the interface in the matching entry is an Ethernet interface other than the receiving interface, the device returns a reply according to the matching entry.
   - If no match is found, the process goes to step 3.

   For information about IPSG, see IP source guard configuration in *Security Configuration Guide*.
3. Searches the ARP snooping table if ARP snooping is enabled on the device. If ARP snooping is disabled on the device, the process goes to step 4.
   - If a match is found and the interface in the matching entry is a wireless interface, the device returns a reply according to the ARP snooping entry.
   - If a match is found and the interface in the matching entry is the Ethernet interface that received the ARP request, the device does not return any reply.
   - If a match is found and the interface in the matching entry is an Ethernet interface other than the receiving interface, the device returns a reply according to the ARP snooping entry.
   - If no match is found, the process goes to step 4.
4. Forwards the ARP request to other interfaces except the receiving interface in the VLAN, or delivers the ARP request to other modules.

Enabling ARP fast-reply

Restrictions and guidelines

To improve the availability of ARP fast-reply, enable ARP snooping at the same time.

Procedure

1. Enter system view.

   system-view
2. Enter VLAN view.

   vlan *vlan-id*
3. Enable ARP fast-reply.

   arp fast-reply enable

By default, ARP fast-reply is disabled.

Configuring ARP direct route advertisement

About ARP direct route advertisement

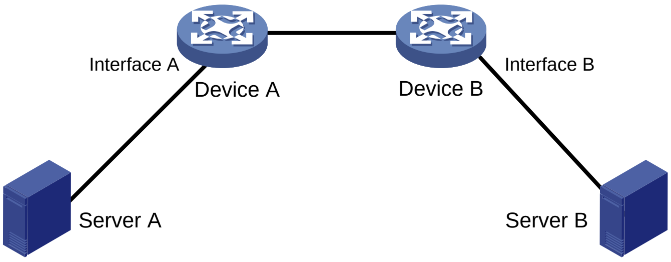
Mechanism of ARP direct route advertisement

The ARP direct route advertisement feature generates host routes based on ARP entries for packet forwarding and route advertisement.

Application in Layer 3 networks

As shown in Figure 3, ARP direct route advertisement is enabled on Interface A and Interface B. This feature advertises a host route to Server A and a host route to Device B, respectively. The routing protocol advertises these host routes rather than the network routes, reducing unexpected traffic caused by network route advertisements.

Figure 3 Application in a Layer 3 network



Enabling ARP direct route advertisement

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable the ARP direct route advertisement feature.

   `arp route-direct advertise`

   By default, the ARP direct route advertisement feature is disabled.

Contents

Configuring basic IPv6 settings

About IPv6

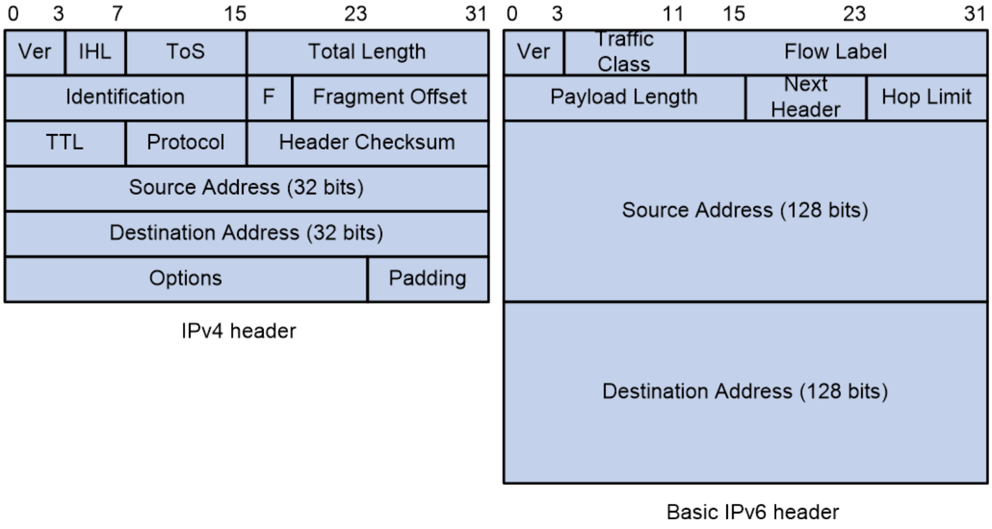
IPv6, also called IP next generation (IPng), was designed by the IETF as the successor to IPv4. One significant difference between IPv6 and IPv4 is that IPv6 increases the IP address size from 32 bits to 128 bits.

IPv6 features

Simplified header format

IPv6 removes several IPv4 header fields or moves them to the IPv6 extension headers to reduce the length of the basic IPv6 packet header. The basic IPv6 packet header has a fixed length of 40 bytes to simplify IPv6 packet handling and improve forwarding efficiency. Although the IPv6 address size is four times the IPv4 address size, the basic IPv6 packet header size is only twice the size of the option-less IPv4 packet header.

Figure 1 IPv4 packet header format and basic IPv6 packet header format



Larger address space

IPv6 can provide 3.4 x 10^{38} addresses to meet the requirements of hierarchical address assignment for both public and private networks.

Hierarchical address structure

IPv6 uses a hierarchical address structure to speed up route lookup and reduce the IPv6 routing table size through route aggregation.

Address autoconfiguration

To simplify host configuration, IPv6 supports stateful and stateless address autoconfiguration.

- Stateful address autoconfiguration enables a host to acquire an IPv6 address and other configuration information from a server (for example, a DHCPv6 server). For more information about DHCPv6 server, see "Configuring the DHCPv6 server."

- Stateless address autoconfiguration enables a host to automatically generate an IPv6 address and other configuration information by using its link-layer address and the prefix information advertised by a router.

To communicate with other hosts on the same link, a host automatically generates a link-local address based on its link-layer address and the link-local address prefix (FE80::/10).

Built-in security

IPv6 defines extension headers to support IPsec. IPsec provides end-to-end security and enhances interoperability among different IPv6 applications.

QoS support

The Flow Label field in the IPv6 header allows the device to label the packets of a specific flow for special handling.

Enhanced neighbor discovery mechanism

The IPv6 neighbor discovery protocol uses a group of ICMPv6 messages to manage information exchange among neighboring nodes on the same link. The group of ICMPv6 messages replaces ARP messages, ICMPv4 router discovery messages, and ICMPv4 redirect messages and provides a series of other functions.

Flexible extension headers

IPv6 eliminates the Options field in the header and introduces optional extension headers to provide scalability and improve efficiency. The Options field in the IPv4 packet header contains a maximum of 40 bytes, whereas the IPv6 extension headers are restricted to the maximum size of IPv6 packets.

IPv6 addresses

IPv6 address format

An IPv6 address is represented as a set of 16-bit hexadecimals separated by colons (:). An IPv6 address is divided into eight groups, and each 16-bit group is represented by four hexadecimal numbers, for example, 2001:0000:130F:0000:0000:09C0:876A:130B.

To simplify the representation of IPv6 addresses, you can handle zeros in IPv6 addresses by using the following methods:

- The leading zeros in each group can be removed. For example, the above address can be represented in a shorter format as 2001:0:130F:0:0:9C0:876A:130B.
- If an IPv6 address contains one or more consecutive groups of zeros, they can be replaced by a double colon (::). For example, the above address can be represented in the shortest format as 2001:0:130F::9C0:876A:130B.

IMPORTANT:

A double colon can appear once or not at all in an IPv6 address. This limit allows the device to determine how many zeros the double colon represents and correctly convert it to zeros to restore a 128-bit IPv6 address.

An IPv6 address consists of an address prefix and an interface ID, which are equivalent to the network ID and the host ID of an IPv4 address.

An IPv6 address prefix is written in IPv6-address/prefix-length notation. The prefix-length is a decimal number indicating how many leftmost bits of the IPv6 address are in the address prefix.

IPv6 address types

IPv6 addresses include the following types:

- **Unicast address**—An identifier for a single interface, similar to an IPv4 unicast address. A packet sent to a unicast address is delivered to the interface identified by that address.

- **Multicast address**—An identifier for a set of interfaces (typically belonging to different nodes), similar to an IPv4 multicast address. A packet sent to a multicast address is delivered to all interfaces identified by that address.

  Broadcast addresses are replaced by multicast addresses in IPv6.
- **Anycast address**—An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to an anycast address is delivered to the nearest interface among the interfaces identified by that address. The nearest interface is chosen according to the routing protocol's measure of distance.

The type of an IPv6 address is designated by the first several bits, called the format prefix.

Table 1 Mappings between address types and format prefixes

Type		Format prefix (binary)	IPv6 prefix ID
Unicast address	Unspecified address	00...0 (128 bits)	::/128
	Loopback address	00...1 (128 bits)	::1/128
	Link-local address	1111111010	FE80::/10
	Global unicast address	Other forms	N/A
Multicast address		11111111	FF00::/8
Anycast address		Anycast addresses use the unicast address space and have the identical structure of unicast addresses.	

Unicast addresses

Unicast addresses include global unicast addresses, link-local unicast addresses, the loopback address, and the unspecified address.

- **Global unicast addresses**—Equivalent to public IPv4 addresses, global unicast addresses are provided for Internet service providers. This type of address allows for prefix aggregation to restrict the number of global routing entries.
- **Link-local addresses**—Used for communication among link-local nodes for neighbor discovery and stateless autoconfiguration. Packets with link-local source or destination addresses are not forwarded to other links.
- **A loopback address**—0:0:0:0:0:0:0:1 (or ::1). It has the same function as the loopback address in IPv4. It cannot be assigned to any physical interface. A node uses this address to send an IPv6 packet to itself.
- **An unspecified address**—0:0:0:0:0:0:0:0 (or ::). It cannot be assigned to any node. Before acquiring a valid IPv6 address, a node fills this address in the source address field of IPv6 packets. The unspecified address cannot be used as a destination IPv6 address.

Multicast addresses

IPv6 multicast addresses listed in Table 2 are reserved for special purposes.

Table 2 Reserved IPv6 multicast addresses

Address	Application
FF01::1	Node-local scope all-nodes multicast address.
FF02::1	Link-local scope all-nodes multicast address.
FF01::2	Node-local scope all-routers multicast address.
FF02::2	Link-local scope all-routers multicast address.

Multicast addresses also include solicited-node addresses. A node uses a solicited-node multicast address to acquire the link-layer address of a neighboring node on the same link and to detect duplicate addresses. Each IPv6 unicast or anycast address has a corresponding solicited-node address. The format of a solicited-node multicast address is FF02:0:0:0:0:1:FFXX:XXXX. FF02:0:0:0:0:1:FF is fixed and consists of 104 bits, and XX:XXXX is the last 24 bits of an IPv6 unicast address or anycast address.

EUI-64 address-based interface identifiers

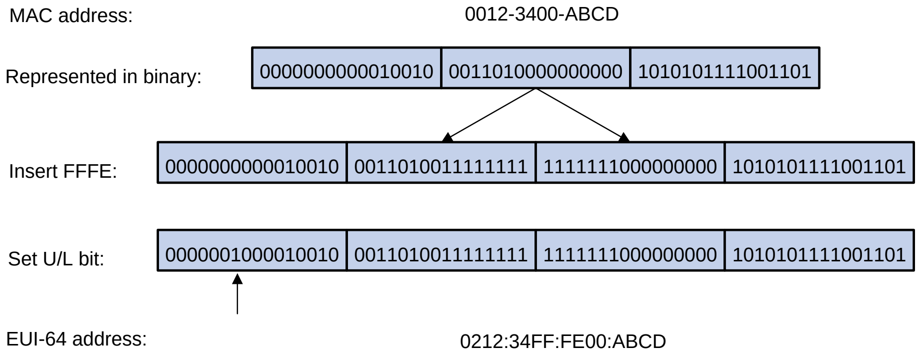
An interface identifier is 64 bits long and uniquely identifies an interface on a link.

On an IEEE 802 interface (such as a VLAN interface), the interface identifier is derived from the link-layer address (typically a MAC address) of the interface. The MAC address is 48 bits long.

To obtain an EUI-64 address-based interface identifier, follow these steps:

1. Insert the 16-bit binary number 1111111111111110 (hexadecimal value of FFFE) behind the 24th high-order bit of the MAC address.
2. Invert the universal/local (U/L) bit (the seventh high-order bit). This operation makes the interface identifier have the same local or global significance as the MAC address.

Figure 2 Converting a MAC address into an EUI-64 address-based interface identifier



On a tunnel interface, the lower 32 bits of the EUI-64 address-based interface identifier are the source IPv4 address of the tunnel interface. The higher 32 bits of the EUI-64 address-based interface identifier of an ISATAP tunnel interface are 0000:5EFE, whereas those of other tunnel interfaces are all zeros. For more information about tunnels, see *VPN Configuration Guide*.

On an interface of another type the EUI-64 address-based interface identifier is generated randomly by the device.

IPv6 path MTU discovery

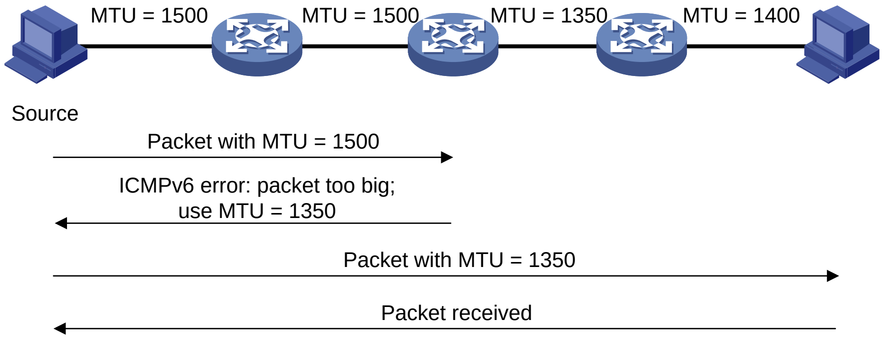
The links that a packet passes from a source to a destination can have different MTUs, among which the minimum MTU is the path MTU. If a packet exceeds the path MTU, the source end fragments the packet to reduce the processing pressure on intermediate devices and to use network resources effectively.

A source end uses path MTU discovery to find the path MTU to a destination, as shown in Figure 3.

1. The source host sends a packet no larger than its MTU to the destination host.
2. If the MTU of an intermediate device's output interface is smaller than the packet, the device performs the following operations:
   - Discards the packet.
   - Returns an ICMPv6 error message containing the interface MTU to the source host.
3. Upon receiving the ICMPv6 error message, the source host performs the following operations:
   - Uses the returned MTU to limit the packet size.
   - Performs fragmentation.
   - Sends the fragments to the destination host.

4. Step 2 and step 3 are repeated until the destination host receives the packet. In this way, the source host finds the minimum MTU of all links in the path to the destination host.

Figure 3 Path MTU discovery process



IPv6 transition technologies

IPv6 transition technologies enable communication between IPv4 and IPv6 networks.

Dual stack

Dual stack is the most direct transition approach. A network node that supports both IPv4 and IPv6 is a dual-stack node. A dual-stack node configured with an IPv4 address and an IPv6 address can forward both IPv4 and IPv6 packets. An application that supports both IPv4 and IPv6 prefers IPv6 at the network layer.

Dual stack is suitable for communication between IPv4 nodes or between IPv6 nodes. It is the basis of all transition technologies. However, it does not solve the IPv4 address depletion issue because each dual-stack node must have a globally unique IPv4 address.

Tunneling

Tunneling uses one network protocol to encapsulate the packets of another network protocol and transfers them over the network. For more information about tunneling, see *VPN Configuration Guide*.

AFT

Address Family Translation (AFT) translates an IP address of one address family into an IP address of the other address family, enabling an IPv4 network and an IPv6 network to communicate with each other. Configured on the edge devices of the IPv4 and IPv6 networks, AFT is transparent to users and does not require configuration changes on IPv4 hosts and IPv6 hosts. For more information about AFT, see *NAT Configuration Guide*.

Protocols and standards

- RFC 1881, *IPv6 Address Allocation Management*
- RFC 1887, *An Architecture for IPv6 Unicast Address Allocation*
- RFC 1981, *Path MTU Discovery for IP version 6*
- RFC 2375, *IPv6 Multicast Address Assignments*
- RFC 2460, *Internet Protocol, Version 6 (IPv6) Specification*
- RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*
- RFC 2526, *Reserved IPv6 Subnet Anycast Addresses*
- RFC 3307, *Allocation Guidelines for IPv6 Multicast Addresses*
- RFC 4191, *Default Router Preferences and More-Specific Routes*

- RFC 4291, *IP Version 6 Addressing Architecture*
- RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*
- RFC 4862, *IPv6 Stateless Address Autoconfiguration*

IPv6 basics tasks at a glance

To configure basic IPv6 settings, perform the following tasks:

1. Configuring an IPv6 address

   Choose the following tasks as needed:
   - Configuring an IPv6 global unicast address
   - Configuring an IPv6 link-local address
   - Configuring an IPv6 anycast address
2. (Optional.) Configuring path MTU discovery
   - Setting the interface MTU
   - Setting a static path MTU for an IPv6 address
   - Setting the aging time for dynamic path MTUs
3. (Optional.) Controlling sending ICMPv6 messages
   - Configuring the rate limit for ICMPv6 error messages
   - Enabling replying to multicast echo requests
   - Enabling sending ICMPv6 destination unreachable messages
   - Enabling sending ICMPv6 time exceeded messages
   - Enabling sending ICMPv6 redirect messages
   - Specifying the source address for ICMPv6 packets
4. (Optional.) Enabling IPv6 local fragment reassembly
5. (Optional.) Enabling discarding IPv6 packets that contain extension headers
6. (Optional.) Configuring IPv6 last hop holding

Configuring an IPv6 global unicast address

About IPv6 global unicast address

Use one of the following methods to configure an IPv6 global unicast address for an interface:

- **EUI-64 IPv6 address**—The IPv6 address prefix of the interface is manually configured, and the interface ID is generated automatically by the interface.
- **Manual configuration**—The IPv6 global unicast address is manually configured.
- **Prefix-specific address autoconfiguration**—The IPv6 global unicast address is generated automatically based on the prefix specified by its ID. The prefix can be manually configured or obtained through DHCPv6.
- **Stateless address autoconfiguration**—The IPv6 global unicast address is generated automatically based on the address prefix information contained in the RA message.

You can configure multiple IPv6 global unicast addresses on an interface.

Manually configured global unicast addresses (including EUI-64 IPv6 addresses) take precedence over automatically generated ones. If you manually configure a global unicast address with the same address prefix as an existing global unicast address on an interface, the manually configured one takes effect. However, it does not overwrite the automatically generated address. If you delete the manually configured global unicast address, the device uses the automatically generated one.

Generating an EUI-64 IPv6 address

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an EUI-64 IPv6 address on the interface.

   ipv6 address { *ipv6-address prefix-length* | *ipv6-address/prefix-length* } **eui-64**

   By default, no EUI-64 IPv6 address is configured on an interface.

Manually assigning an IPv6 global unicast address

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Assign an IPv6 global unicast address to the interface.

   ipv6 address { *ipv6-address prefix-length* | *ipv6-address/prefix-length* }

   By default, no IPv6 global unicast address is configured on an interface.

Stateless address autoconfiguration

About this task

Stateless address autoconfiguration enables an interface to automatically generate an IPv6 global unicast address by using the address prefix in the received RA message and the interface ID. On an IEEE 802 interface (such as an Ethernet interface or a VLAN interface), the interface ID is generated based on the interface's MAC address and is globally unique. An attacker can exploit this rule to identify the sending device easily.

To fix the vulnerability, you can configure the temporary address feature. With this feature, an IEEE 802 interface generates the following addresses:

- **Public IPv6 address**—Includes the address prefix in the RA message and a fixed interface ID generated based on the MAC address of the interface.
- **Temporary IPv6 address**—Includes the address prefix in the RA message and a random interface ID generated through MD5.

You can also configure the interface to preferentially use the temporary IPv6 address as the source address of sent packets. When the valid lifetime of the temporary IPv6 address expires, the interface deletes the address and generates a new one. This feature enables the system to send packets with different source addresses through the same interface. If the temporary IPv6 address cannot be used because of a DAD conflict, the public IPv6 address is used.

The preferred lifetime and valid lifetime for a temporary IPv6 address are determined as follows:

- The preferred lifetime of a temporary IPv6 address takes the smaller of the following values:
  - The preferred lifetime of the address prefix in the RA message.

    - The preferred lifetime configured for temporary IPv6 addresses minus DESYNC_FACTOR (a random number in the range of 0 to 600 seconds).
- The valid lifetime of a temporary IPv6 address takes the smaller of the following values:
    - The valid lifetime of the address prefix.
    - The valid lifetime configured for temporary IPv6 addresses.

Restrictions and guidelines

If the IPv6 prefix in the RA message is not 64 bits long, stateless address autoconfiguration fails to generate an IPv6 global unicast address.

To generate a temporary address, an interface must be enabled with stateless address autoconfiguration. Temporary IPv6 addresses do not overwrite public IPv6 addresses, so an interface can have multiple IPv6 addresses with the same address prefix but different interface IDs.

If an interface fails to generate a public IPv6 address because of a prefix conflict or other reasons, it does not generate any temporary IPv6 address.

Executing the **undo ipv6 address auto** command on an interface deletes all IPv6 global unicast addresses and link-local addresses that are automatically generated on the interface.

Enabling stateless address autoconfiguration

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Enable stateless address autoconfiguration on an interface, so that the interface can automatically generate a global unicast address.
   ipv6 address auto
   By default, the stateless address autoconfiguration feature is disabled on an interface.

Configuring the temporary address feature and preferentially using the temporary IPv6 address as the source address of outgoing packets

1. Enter system view.
   system-view
2. Enable the temporary IPv6 address feature.
   ipv6 temporary-address [*valid-lifetime preferred-lifetime*]
   By default, the temporary IPv6 address feature is disabled.
3. Enable the system to preferentially use the temporary IPv6 address as the source address of the outgoing packets.
   ipv6 prefer temporary-address
   By default, the system does not preferentially use the temporary IPv6 address as the source address of the outgoing packets.

Configuring prefix-specific address autoconfiguration

1. Enter system view.
   system-view
2. Configure an IPv6 prefix.
   Choose one option as needed:
    - Configure a static IPv6 prefix.
      ipv6 prefix *prefix-number ipv6-prefix*/*prefix-length*

By default, no static IPv6 prefixes exist.

  - Use DHCPv6 to obtain a dynamic IPv6 prefix.

    For more information about IPv6 prefix acquisition, see "Configuring the DHCPv6 client."

3. Enter interface view.

   `interface` *`interface-type interface-number`*

4. Specify an IPv6 prefix for an interface to automatically generate an IPv6 global unicast address and advertise the prefix.

   `ipv6 address` *`prefix-number sub-prefix/prefix-length`*

   By default, no IPv6 prefix is specified for the interface to automatically generate an IPv6 global unicast address.

Configuring an IPv6 link-local address

About IPv6 link-local address

Configure IPv6 link-local addresses using one of the following methods:

- **Automatic generation**—The device automatically generates a link-local address for an interface according to the link-local address prefix (FE80::/10) and the link-layer address of the interface.
- **Manual assignment**—Manually configure an IPv6 link-local address for an interface.

Restrictions and guidelines

After you configure an IPv6 global unicast address for an interface, the interface automatically generates a link-local address. This link-local address is the same as the one generated by using the **`ipv6 address auto link-local`** command. If a link-local address is manually assigned to an interface, this manual assigned link-local address takes effect. If the manually assigned link-local address is deleted, the automatically generated link-local address takes effect.

Using the **`undo ipv6 address auto link-local`** command on an interface deletes only the link-local address generated by the **`ipv6 address auto link-local`** command. If the interface has an IPv6 global unicast address, it still has a link-local address. If the interface has no IPv6 global unicast address, it has no link-local address.

An interface can have only one link-local address. As a best practice, use the automatic generation method to avoid link-local address conflicts. If both the automatic generation and manual assignment methods are used, the manual assignment takes precedence.

- If you first use automatic generation and then manual assignment, the manually assigned link-local address overwrites the automatically generated one.
- If you first use manual assignment and then automatic generation, both of the following occur:
  - The link-local address is still the manually assigned one.
  - The automatically generated link-local address does not take effect. If you delete the manually assigned address, the automatically generated link-local address takes effect.

Configuring automatic generation of an IPv6 link-local address for an interface

1. Enter system view.

   `system-view`

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface to automatically generate an IPv6 link-local address.

   ipv6 address auto link-local

   By default, no link-local address is configured on an interface.

   After an IPv6 global unicast address is configured on the interface, a link-local address is generated automatically.

Manually assigning an IPv6 link-local address to an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Manually assign an IPv6 link-local address to the interface.

   ipv6 address *ipv6-address* **link-local**

   By default, no link-local address is configured on an interface.

Configuring an IPv6 anycast address

4. Enter system view.

   system-view

5. Enter interface view.

   interface *interface-type interface-number*

6. Configure an IPv6 anycast address.

   ipv6 address { *ipv6-address prefix-length* | *ipv6-address/prefix-length* } **anycast**

   By default, no IPv6 anycast address is configured on an interface.

Configuring path MTU discovery

Setting the interface MTU

About this task

IPv6 routers do not support packet fragmentation. If a packet exceeds the MTU of the output interface, the router discards the packet and sends a packet too big message to the source host. This message contains the interface MTU. The source host fragments the packet according to the returned MTU. To avoid traffic overload due to packet dropping, set a proper interface MTU.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the interface MTU.

   ipv6 mtu *size*

By default, no interface MTU is set.

Setting a static path MTU for an IPv6 address

About this task

You can set a static path MTU for an IPv6 address. Before sending a packet to the IPv6 address, the device compares the output interface MTU with the static path MTU. If the packet size exceeds the smaller one of the two values, the device fragments the packet according to the smaller value. After sending the fragmented packets, the device dynamically finds the path MTU to a destination host (see "IPv6 path MTU discovery").

Procedure

1. Enter system view.

   system-view

2. Set a static path MTU for an IPv6 address.

   ipv6 pathmtu [**vpn-instance** *vpn-instance-name*] *ipv6-address value*

   By default, no path MTU is set for any IPv6 address.

Setting the aging time for dynamic path MTUs

About this task

After the device dynamically discovers the path MTU to a destination host (see "IPv6 path MTU discovery"), it performs the following operations:

- Sends packets to the destination host based on this path MTU.
- Starts the aging timer for this path MTU.

When the aging timer expires, the device removes the dynamic path MTU and discovers the path MTU again.

Restrictions and guidelines

The aging time is invalid for a static path MTU.

Procedure

1. Enter system view.

   system-view

2. Set the aging time for dynamic path MTUs.

   ipv6 pathmtu age *age-time*

   The default setting is 10 minutes.

Controlling sending ICMPv6 messages

Configuring the rate limit for ICMPv6 error messages

About this task

To avoid sending excessive ICMPv6 error messages within a short period that might cause network congestion, you can limit the rate at which ICMPv6 error messages are sent. A token bucket algorithm is used with one token representing one ICMPv6 error message.

A token is placed in the bucket at intervals until the maximum number of tokens that the bucket can hold is reached.

A token is removed from the bucket when an ICMPv6 error message is sent. When the bucket is empty, ICMPv6 error messages are not sent until a new token is placed in the bucket.

Procedure

1. Enter system view.
   `system-view`
2. Set the bucket size and the interval for tokens to arrive in the bucket for ICMPv6 error messages.
   `ipv6 icmpv6 error-interval` *`interval`* `[` *`bucketsize`* `]`
   By default, the bucket allows a maximum of 10 tokens. A token is placed in the bucket at an interval of 100 milliseconds.
   To disable the ICMPv6 rate limit, set the interval to 0 milliseconds.

Enabling replying to multicast echo requests

1. Enter system view.
   `system-view`
2. Enable replying to multicast echo requests.
   `ipv6 icmpv6 multicast-echo-reply enable`
   By default, this feature is disabled.

Enabling sending ICMPv6 destination unreachable messages

About this task

The device sends the source the following ICMPv6 destination unreachable messages:

- **ICMPv6 No Route to Destination message**—A packet to be forwarded does not match any route.
- **ICMPv6 Communication with Destination Administratively Prohibited message**—An administrative prohibition is preventing successful communication with the destination. This is typically caused by a firewall or an ACL on the device.
- **ICMPv6 Beyond Scope of Source Address message**—The destination is beyond the scope of the source IPv6 address. For example, a packet's source IPv6 address is a link-local address, and its destination IPv6 address is a global unicast address.
- **ICMPv6 Address Unreachable message**—The device fails to resolve the link layer address for the destination IPv6 address of a packet.
- **ICMPv6 Port Unreachable message**—No port process on the destination device exists for a received UDP packet.

Restrictions and guidelines

An ICMPv6 destination unreachable message indicates that the destination is not reachable from the source device. Attackers can launch malicious attacks to make the device generate incorrect ICMPv6 destination unreachable messages, which will affect the function of the network. To protect the network from malicious attacks and decrease unnecessary network traffic, you can disable the sending of ICMPv6 destination unreachable messages.

Procedure

1. Enter system view.
   `system-view`

2. Enable sending ICMPv6 destination unreachable messages.

   `ipv6 unreachables enable`

   By default, this feature is disabled.

Enabling sending ICMPv6 time exceeded messages

About this task

The device sends the source ICMPv6 time exceeded messages as follows:

- If a received packet is not destined for the device and its hop limit is 1, the device sends an ICMPv6 hop limit exceeded in transit message to the source.
- Upon receiving the first fragment of an IPv6 datagram destined for the device, the device starts a timer. If the timer expires before all fragments arrive, the device sends an ICMPv6 fragment reassembly time exceeded message to the source.

Restrictions and guidelines

If the device receives large numbers of malicious packets, its performance degrades greatly because it must send back ICMP time exceeded messages. To prevent such attacks, disable sending ICMPv6 time exceeded messages.

Procedure

1. Enter system view.

   `system-view`

2. Enable sending ICMPv6 time exceeded messages.

   `ipv6 hoplimit-expires enable`

   The default setting is disabled.

Enabling sending ICMPv6 redirect messages

About this task

Upon receiving a packet from a host, the device sends an ICMPv6 redirect message to inform the host of a better next hop when the following conditions are met:

- The interface receiving the packet is the interface forwarding the packet.
- The selected route is not created or modified by any ICMPv6 redirect messages.
- The selected route is not a default route.
- The forwarded packet does not contain the routing extension header.

The ICMPv6 redirect feature simplifies host management by enabling hosts that hold few routes to optimize their routing table gradually. However, to avoid adding too many routes on hosts, this feature is disabled by default.

Procedure

1. Enter system view.

   `system-view`

2. Enable sending ICMPv6 redirect messages.

   `ipv6 redirects enable`

   By default, sending ICMPv6 redirect messages is disabled.

Specifying the source address for ICMPv6 packets

About this task

Perform this task to specify the source IPv6 address for outgoing ping echo requests and ICMPv6 error messages. It is a good practice to specify the IPv6 address of the loopback interface as the source IPv6 address. This feature helps users to easily locate the sending device.

Restrictions and guidelines

If you specify an IPv6 address in the **ping** command, ping echo requests use the specified address as the source IPv6 address. If you do not specify an IPv6 address in the **ping** command, ping echo requests use the IPv6 address specified by the **ipv6 icmpv6 source** command.

Procedure

1. Enter system view.

   system-view

2. Specify an IPv6 address as the source address for outgoing ICMPv6 packets.

   ipv6 icmpv6 source [**vpn-instance** *vpn-instance-name*] *ipv6-address*

   By default, the device uses the IPv6 address of the sending interface as the source IPv6 address for outgoing ICMPv6 packets.

Enabling IPv6 local fragment reassembly

About this task

Use this feature on a multichassis IRF fabric to improve fragment reassembly efficiency. This feature enables a subordinate to reassemble the IPv6 fragments of a packet if all the fragments arrive at it. If this feature is disabled, all IPv6 fragments are delivered to the master device for reassembly.

Restrictions and guidelines

The IPv6 local fragment reassembly feature applies only to fragments destined for the same subordinate.

Procedure

1. Enter system view.

   system-view

2. Enable IPv6 local fragment reassembly.

   ipv6 reassemble local enable

   By default, IPv6 local fragment reassembly is disabled.

Enabling discarding IPv6 packets that contain extension headers

About this task

This feature enables a device to discard a received IPv6 packet in which the extension headers cannot be processed by the device.

Procedure

1. Enter system view

   system-view

2. Enable the device to discard IPv6 packets that contain extension headers.

   ipv6 extension-header drop enable

   By default, the device does not discard IPv6 packets that contain extension headers.

Configuring IPv6 last hop holding

About this task

Last hop holding implements symmetric routing.

When the interface enabled with this feature receives the first IPv6 packet of a forward flow, this feature implements the following operations:

- Obtains the forward flow information and last hop information of the packet.
- Based on the information, creates an IPv6 fast forwarding entry for the reverse flow.

When packets of the reverse flow arrive at the device, the device forwards those packets based on the entry.

Restrictions and guidelines

Last hop holding is based on IPv6 fast forwarding entries. If the MAC address of a last hop changes on an Ethernet link, this feature can function correctly only after the fast forwarding entry is updated for the MAC address.

Procedure

1. Enter system view.

   system-view

2. Enter interface view. Choose one option as needed:
   - Enter Layer 3 Ethernet interface view or Layer 3 Ethernet subinterface view.

     interface *interface-type interface-number*
   - Enter dialer interface view.

     interface dialer *number*

3. Enable IPv6 last hop holding.

   ipv6 last-hop hold

   By default, IPv6 last hop holding is disabled.

Display and maintenance commands for IPv6 basics

Execute **display** commands in any view and **reset** commands in user view.

For information about the **display tcp statistics**, **display udp statistics**, **reset tcp statistics**, and **reset udp statistics** command, see the IP performance commands in *Layer 3—IP Services Command Reference*.

Task	Command
Display IPv6 FIB entries.	**display ipv6 fib** [**vpn-instance** *vpn-instance-name*] [*ipv6-address* [*prefix-length*]]
Display ICMPv6 traffic statistics.	**display ipv6 icmp statistics** [**slot**

Task	Command
	slot-number]
Display IPv6 information about the interface.	**display ipv6 interface** [*interface-type* [*interface-number*]] [**brief**]
Display IPv6 prefix information about the interface.	**display ipv6 interface** *interface-type interface-number* **prefix**
Display the IPv6 path MTU information.	**display ipv6 pathmtu** [**vpn-instance** *vpn-instance-name*] { *ipv6-address* | { **all** | **dynamic** | **static** } [**count**] }
Display the IPv6 prefix information.	**display ipv6 prefix** [*prefix-number*]
Display brief information about IPv6 RawIP connections.	**display ipv6 rawip** [**slot** *slot-number*]
Display detailed information about IPv6 RawIP connections.	**display ipv6 rawip verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Display IPv6 and ICMPv6 packet statistics.	**display ipv6 statistics** [**slot** *slot-number*]
Display brief information about IPv6 TCP connections.	**display ipv6 tcp** [**slot** *slot-number*]
Display detailed information about IPv6 TCP connections.	**display ipv6 tcp verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Display brief information about IPv6 TCP proxy.	**display ipv6 tcp-proxy slot** *slot-number*
Display the usage of non-well known ports for IPv6 TCP proxy.	**display ipv6 tcp-proxy port-info slot** *slot-number*
Display brief information about IPv6 UDP connections.	**display ipv6 udp** [**slot** *slot-number*]
Display detailed information about IPv6 UDP connections.	**display ipv6 udp verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Display IPv6 TCP traffic statistics.	**display tcp statistics** [**slot** *slot-number*]
Display IPv6 UDP traffic statistics.	**display udp statistics** [**slot** *slot-number*]
Clear path MTUs.	**reset ipv6 pathmtu** { **all** | **dynamic** | **static** }
Clear IPv6 and ICMPv6 packet statistics.	**reset ipv6 statistics** [**slot** *slot-number*]
Clear IPv6 TCP traffic statistics.	**reset tcp statistics**
Clear IPv6 UDP traffic statistics.	**reset udp statistics**

Configuring IPv6 neighbor discovery

About IPv6 neighbor discovery

ICMPv6 messages used by IPv6 neighbor discovery

The IPv6 neighbor discovery (ND) process uses ICMP messages for address resolution, neighbor reachability verification, and neighboring device tracking.

Table 3 describes the ICMPv6 messages used by the IPv6 ND protocol.

Table 3 ICMPv6 messages used by ND

ICMPv6 message	Type	Function
Neighbor Solicitation (NS)	135	Acquires the link-layer address of a neighbor on the local link.
		Verifies the reachability of a neighbor.
		Detects duplicate addresses.
Neighbor Advertisement (NA)	136	Responds to an NS message.
		Notifies the neighboring nodes of link layer changes.
Router Solicitation (RS)	133	Requests an address prefix and other configuration information for autoconfiguration after startup.
Router Advertisement (RA)	134	Responds to an RS message.
		Advertises information, such as the Prefix Information options and flag bits.
Redirect	137	Informs the source host of a better next hop on the path to a particular destination when certain conditions are met.

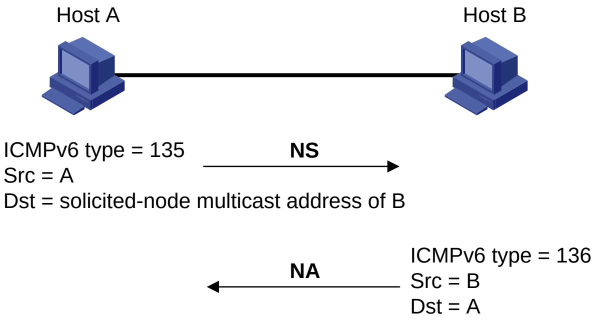
Address resolution

This function is similar to ARP in IPv4. An IPv6 node acquires the link-layer addresses of neighboring nodes on the same link through NS and NA messages.

Figure 4 shows how Host A acquires the link-layer address of Host B on the same link. The address resolution procedure is as follows:

1. Host A multicasts an NS message. The source address of the NS message is the IPv6 address of the sending interface of Host A. The destination address is the solicited-node multicast address of Host B. The NS message body contains the link-layer address of Host A and the target IPv6 address.
2. After receiving the NS message, Host B determines whether the target address of the packet is its IPv6 address. If it is, Host B learns the link-layer address of Host A, and then unicasts an NA message containing its link-layer address.
3. Host A acquires the link-layer address of Host B from the NA message.

Figure 4 Address resolution



Neighbor reachability detection

After Host A acquires the link-layer address of its neighbor Host B, Host A can use NS and NA messages to test the reachability of Host B as follows:

1. Host A sends an NS message whose destination address is the IPv6 address of Host B.
2. If Host A receives an NA message from Host B, Host A decides that Host B is reachable. Otherwise, Host B is unreachable.

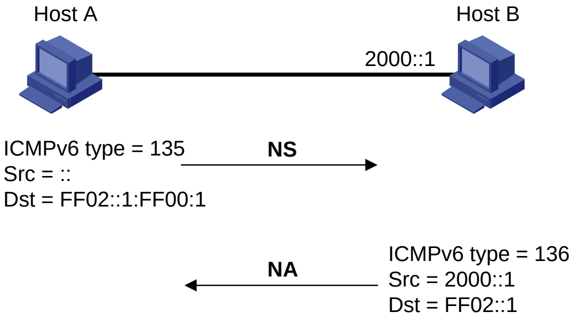
Duplicate address detection

After Host A acquires an IPv6 address, it performs Duplicate Address Detection (DAD) to check whether the address is being used by any other node. This is similar to gratuitous ARP in IPv4. DAD is accomplished through NS and NA messages.

The DAD procedure is as follows:

1. Host A sends an NS message. The source address is the unspecified address and the destination address is the corresponding solicited-node multicast address of the IPv6 address to be detected. The NS message body contains the detected IPv6 address.
2. If Host B uses this IPv6 address, Host B returns an NA message that contains its IPv6 address.
3. Host A knows that the IPv6 address is being used by Host B after receiving the NA message from Host B. If receiving no NA message, Host A decides that the IPv6 address is not in use and uses this address.

Figure 5 Duplicate address detection



Router/prefix discovery and stateless address autoconfiguration

Router/prefix discovery allows an IPv6 node to find the neighboring routers and learn the prefix and network configuration parameters of the network from receiving RA messages.

Stateless address autoconfiguration allows an IPv6 node to automatically generate an IPv6 address based on the information learned through router/prefix discovery.

A node performs router/prefix discovery and stateless address autoconfiguration as follows:

1. At startup, a node sends an RS message to request configuration information from a router.
2. The router returns an RA message containing the Prefix Information option and other configuration information. (The router also periodically sends an RA message.)
3. The node automatically generates an IPv6 address and other configuration parameters according to the configuration information in the RA message.

The Prefix Information option contains an address prefix and the preferred lifetime and valid lifetime of the address prefix. A node updates the preferred lifetime and valid lifetime upon receiving a periodic RA message.

The generated IPv6 address is valid within the valid lifetime and becomes invalid when the valid lifetime expires.

After the preferred lifetime expires, the node cannot use the generated IPv6 address to establish new connections, but can receive packets destined for the IPv6 address. The preferred lifetime cannot be greater than the valid lifetime.

Redirection

Upon receiving a packet from a host, the gateway sends an ICMPv6 redirect message to inform the host of a better next hop when the following conditions are met:

- The interface receiving the packet is the same as the interface forwarding the packet.
- The selected route is not created or modified by an ICMPv6 redirect message.
- The selected route is not a default route on the device.
- The forwarded IPv6 packet does not contain the routing extension header.

Protocols and standards

- RFC 4861, *Neighbor Discovery for IP Version 6 (IPv6)*
- RFC 8106, *IPv6 Router Advertisement Options for DNS Configuration*

IPv6 neighbor discovery tasks at a glance

All IPv6 neighbor discovery tasks are optional.

- Configuring a static neighbor entry
- Setting the dynamic neighbor learning limit on an interface
- Enabling unsolicited NA learning
- Setting the aging timer for ND entries in stale state
- Minimizing link-local ND entries
- Setting the hop limit
- Configuring RA message sending and parameters

Configuring a static neighbor entry

About this task

A neighbor entry stores information about a link-local node. The entry can be created dynamically through NS and NA messages, or configured statically.

The device uniquely identifies a static neighbor entry by using the neighbor's IPv6 address and the number of the Layer 3 interface that connects to the neighbor. You can configure a static neighbor entry by using one of the following methods:

- **Method 1**—Associate a neighbor's IPv6 address and link-layer address with the local Layer 3 interface.
- **Method 2**—Associate a neighbor's IPv6 address and link-layer address with a Layer 2 port in a VLAN.

Restrictions and guidelines

You can use either of the methods to configure a static neighbor entry for a VLAN interface.

- If you use Method 1, the device is required to resolve the Layer 2 port in the related VLAN.
- If you use Method 2, make sure the Layer 2 port belongs to the specified VLAN and the corresponding VLAN interface already exists. After the configuration, the device associates the VLAN interface with the neighbor IPv6 address to identify the static neighbor entry.

Do not specify a Reth interface as the outgoing interface in IPv6 static neighbor entries if its member interfaces contain subinterfaces. For more information about Reth interfaces, see *Virtual Technologies Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Configure a static neighbor entry.

   `ipv6 neighbor` *`ipv6-address mac-address`* `{` *`vlan-id port-type port-number`* `|` **`interface`** *`interface-type interface-number`* `}` `[` **`vpn-instance`** *`vpn-instance-name`* `]`

   By default, no static neighbor entries exist.

Setting the dynamic neighbor learning limit on an interface

About this task

The device can dynamically acquire the link-layer address of a neighboring node through NS and NA messages and add it into the neighbor table. When the number of dynamic neighbor entries reaches the limit, the interface stops learning neighbor information.

This feature limits the neighbor table size. A large neighbor table will degrade the forwarding performance.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the dynamic neighbor learning limit on the interface.

   ipv6 neighbors max-learning-num *max-number*

   By default, an interface can learn a maximum of 65536 dynamic neighbor entries.

Enabling unsolicited NA learning

About this task

On some networks, a server multicasts NA messages to two peer devices for link backup. The peer devices cannot learn ND entry for the server from these NA messages by default. If no ND learning is triggered by data exchange between the server and peer devices, the peer devices learn the entry for the server only when the server unicasts messages to them.

Restrictions and guidelines

This feature enables an interface to learn ND entries from unsolicited NA messages. The ND entries generated by using this method are in stale state. To ensure that the device learns ND entries from trusted NA messages, enable this feature only on a secure network.

This feature might cause the device to learn excessive ND entries that consume too many system resources. As a best practice, execute the **ipv6 neighbor stale-aging** command to set a smaller aging timer before you enable this feature. The smaller aging timer accelerates the aging of ND entries in stale state.

This feature is available only on Layer 3 interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter Layer 3 interface view.

   interface *interface-type interface-number*

3. Enable unsolicited NA learning.

   ipv6 nd unsolicited-na-learning enable

   By default, unsolicited NA learning is disabled.

Setting the aging timer for ND entries in stale state

About this task

ND entries in stale state have an aging timer. If an ND entry in stale state is not refreshed before the timer expires, the ND entry changes to the delay state. If it is still not refreshed in 5 seconds, the ND entry changes to the probe state, and the device sends an NS message three times. If no response is received, the device deletes the ND entry.

Procedure

1. Enter system view.

   system-view

2. Set the aging timer for ND entries in stale state.

```
ipv6 neighbor stale-aging aging-time
```

The default setting is 240 minutes.

Minimizing link-local ND entries

About this task

Perform this task to minimize link-local ND entries assigned to the hardware. Link-local ND entries refer to ND entries that contain link-local addresses.

By default, the device assigns all ND entries to the hardware. With this feature enabled, the newly learned link-local ND entries are not assigned to the hardware if the link-local addresses of the entries are not the next hops of any routes. This feature saves hardware resources.

This feature takes effect only on newly learned link-local ND entries.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Minimize link-local ND entries.

   ```
   ipv6 neighbor link-local minimize
   ```

   By default, the device assigns all ND entries to the hardware.

Setting the hop limit

About this task

You can set the hop limit value to fill in the Hop Limit field for IPv6 packets to be sent.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the value for the Hop Limit field in the IP header.

   ```
   ipv6 hop-limit value
   ```

   The default setting is 64.

Configuring RA message sending and parameters

About RA message parameters

You can enable an interface to send RA messages, and configure the interval for sending RA messages and parameters in RA messages. After receiving an RA message, a host can use these parameters to perform corresponding operations. Table 4 describes the configurable parameters in an RA message.

Table 4 Parameters in an RA message and their descriptions

Parameter	Description
Hop Limit	Maximum number of hops in RA messages. A host receiving the RA message fills the value in the Hop Limit field of sent IPv6 packets.
Prefix information	After receiving the prefix information, the hosts on the same link can perform stateless autoconfiguration.

Parameter	Description
MTU	Guarantees that all nodes on the link use the same MTU.
M flag	Determines whether a host uses stateful autoconfiguration to obtain an IPv6 address. If the M flag is set to 1, the host uses stateful autoconfiguration (for example, from a DHCPv6 server) to obtain an IPv6 address. Otherwise, the host uses stateless autoconfiguration to generate an IPv6 address according to its link-layer address and the prefix information in the RA message.
O flag	Determines whether a host uses stateful autoconfiguration to obtain configuration information other than the IPv6 address. If the O flag is set to 1, the host uses stateful autoconfiguration (for example, from a DHCPv6 server) to obtain configuration information other than the IPv6 address. Otherwise, the host uses stateless autoconfiguration.
Router Lifetime	Tells the receiving hosts how long the advertising router can live. If the lifetime of a router is 0, the router cannot be used as the default gateway.
Retrans Timer	If the device does not receive a response message within the specified time after sending an NS message, it retransmits the NS message.
Reachable Time	If the neighbor reachability detection shows that a neighbor is reachable, the device considers the neighbor reachable within the specified reachable time. If the device needs to send a packet to the neighbor after the specified reachable time expires, the device reconfirms whether the neighbor is reachable.
Router Preference	Specifies the router preference in an RA message. A host selects a router as the default gateway according to the router preference. If router preferences are the same, the host selects the router from which the first RA message is received.
DNS server option	DNS server information for IPv6 hosts. Hosts can obtain DNS server information from received RA messages instead of using DHCPv6.
DNS suffix information in DNS Search List (DNSSL) option	DNS suffix information for IPv6 hosts. Hosts can obtain DNS suffix information from received RA messages instead of using DHCPv6.

Restrictions and guidelines

The maximum interval for sending RA messages should be less than (or equal to) the router lifetime in RA messages. In this way, the router can be updated by an RA message before expiration.

The values of the NS retransmission timer and the reachable time configured for an interface are sent in RA messages to hosts. This interface sends NS messages at the interval of the NS retransmission timer and considers a neighbor reachable within the reachable time.

Enabling the sending of RA messages

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Enable the sending of RA messages.
   undo ipv6 nd ra halt
   The default setting is disabled.
4. Set the maximum and minimum intervals for sending RA messages.
   ipv6 nd ra interval *max-interval min-interval*

By default, the maximum interval for sending RA messages is 600 seconds, and the minimum interval is 200 seconds.

The device sends RA messages at random intervals between the maximum interval and the minimum interval.

The minimum interval should be less than or equal to 0.75 times the maximum interval.

Configuring parameters for RA messages

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the prefix information in RA messages.

   ipv6 nd ra prefix { *ipv6-prefix prefix-length* | *ipv6-prefix***/***prefix-length* } [*valid-lifetime preferred-lifetime* [**no-autoconfig** | **off-link**] * | **no-advertise**]

   By default, no prefix information is configured for RA messages, and the IPv6 address of the interface sending RA messages is used as the prefix information. If the IPv6 address is manually configured, the prefix uses a fixed valid lifetime of 2592000 seconds (30 days) and a preferred lifetime of 604800 seconds (7 days). If the IPv6 address is automatically obtained (through DHCP, for example), the prefix uses the valid lifetime and preferred lifetime configured for the IPv6 address.

4. Configure the default settings for prefixes advertised in RA messages.

   ipv6 nd ra prefix default [*valid-lifetime preferred-lifetime* [**no-autoconfig** | **off-link**] * | **no-advertise**]

   By default, no default settings are configured for prefixes advertised in RA messages.

5. Turn off the MTU option in RA messages.

   ipv6 nd ra no-advlinkmtu

   By default, RA messages contain the MTU option.

6. Specify unlimited hops in RA messages.

   ipv6 nd ra hop-limit unspecified

   By default, the maximum number of hops in RA messages is 64.

7. Set the M flag bit to 1.

   ipv6 nd autoconfig managed-address-flag

   By default, the M flag bit is set to 0 in RA advertisements. Hosts receiving the advertisements will obtain IPv6 addresses through stateless autoconfiguration.

8. Set the O flag bit to 1.

   ipv6 nd autoconfig other-flag

   By default, the O flag bit is set to 0 in RA advertisements. Hosts receiving the advertisements will acquire other configuration information through stateless autoconfiguration.

9. Set the router lifetime in RA messages.

   ipv6 nd ra router-lifetime *time*

   By default, the router lifetime is three times as long as the maximum interval for advertising RA messages.

10. Set the NS retransmission timer.

    ipv6 nd ns retrans-timer *value*

By default, an interface sends NS messages every 1000 milliseconds, and the value of the Retrans Timer field in RA messages is 0.

11. Set the router preference in RA messages.

```
ipv6 nd router-preference { high | low | medium }
```

By default, the router preference is medium.

12. Set the reachable time.

```
ipv6 nd nud reachable-time time
```

By default, the neighbor reachable time is 30000 milliseconds, and the value of the Reachable Time field in sent RA messages is 0.

Specifying DNS server information in RA messages

About this task

The DNS server options in RA messages provide DNS server information for IPv6 hosts. The RA messages allow hosts to obtain their IPv6 addresses and the DNS server through stateless autoconfiguration. This method is useful in a network where DHCPv6 infrastructure is not provided.

One DNS server option contains one DNS server. All DNS server options are sorted in ascending order of the DNS server sequence number.

After you execute the **ipv6 nd ra dns server** command, the device immediately sends an RA message with the existing and newly specified DNS server information.

After you execute the **undo ipv6 nd ra dns server** command, the device immediately sends two RA messages.

- The first RA message contains information about all DNS servers, including the DNS servers specified in the **undo** command with their lifetime set to 0 seconds.
- The second RA message contains information about remaining DNS servers.

Each time the device sends an RA message from an interface, it immediately refreshes the RA message advertisement interval for that interface.

Restrictions and guidelines

You can configure a maximum of eight DNS servers on an interface.

The default lifetime of a DNS server is three times the maximum interval for advertising RA messages. To set the maximum interval, use the **ipv6 nd ra interval** command.

In an IPv6 environment, PPP users and IPoE IPv6-ND-RS users can obtain the IPv6 DNS server address through AAA authorization. This AAA-authorized IPv6 DNS server address is also carried in RA messages. If an interface obtains the AAA-authorized and manually specified IPv6 DNS server addresses, the RA messages contain both, with the AAA-authorized address in the front. When the two addresses conflict, the AAA-authorized DNS-related attributes are used.

For more information about the PPP support for IPv6, see PPP configuration in *Layer 2—WAN Access Configuration Guide*.

For more information about IPoE IPv6-ND-RS users, see IPoE configuration in *Security Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Specify DNS server information to be advertised in RA messages.

ipv6 nd ra dns server *ipv6-address* [*seconds* | **infinite**] **sequence** *seqno*

By default, no DNS server information is specified and RA messages do not contain DNS server options.

Specifying DNS suffix information in RA messages

About this task

The DNSSL option in RA messages provides suffix information for IPv6 hosts. The RA messages allow hosts to obtain their IPv6 addresses and the DNS suffix through stateless autoconfiguration. This method is useful in a network where DHCPv6 infrastructure is not provided.

One DNSSL option contains one DNS suffix. All DNSSL options are sorted in ascending order of the sequence number of the DNS suffix.

After you execute the **ipv6 nd ra dns search-list** command, the device immediately sends an RA message with the existing and newly specified DNS suffix information.

After you execute the **undo ipv6 nd ra dns search-list** command, the device immediately sends two RA messages.

- The first RA message contains information about all DNS suffixes, including DNS suffixes specified in the **undo** command with their lifetime set to 0 seconds.
- The second RA message contains information about remaining DNS suffixes.

Each time the device sends an RA message from an interface, it immediately refreshes the RA message advertisement interval for that interface.

Restrictions and guidelines

You can configure a maximum of eight DNS suffixes on an interface.

The default lifetime of a DNS suffix is three times the maximum interval for advertising RA messages. To set the maximum interval, use the **ipv6 nd ra interval** command.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify DNS suffix information to be advertised in RA messages.

   ipv6 nd ra dns search-list *domain-name* [*seconds* | **infinite**] **sequence** *seqno*

   By default, no DNS suffix information is specified and RA messages do not contain DNS suffix options.

Suppressing advertising DNS information in RA messages

About this task

Perform this task to suppress the device from advertising information about DNS server addresses and DNS suffixes in RA messages.

Whether enabling this feature on an interface will trigger sending RA message immediately depends on the interface configuration:

- If the interface has DNS server information configured or has obtained an AAA-authorized DNS server address, the device immediately sends two RA messages. In the first message, the lifetime for DNS server addresses is 0 seconds. The second RA message does not contain any DNS server options.

- If the interface has no DNS server information specified or no AAA-authorized DNS server address assigned, no RA messages are triggered.

If you specify a new DNS server or remove a DNS server on the interface after enabling DNS server suppression, the device immediately sends an RA message without any DNS server options.

Whether disabling this feature on an interface will trigger sending RA message immediately depends on the interface configuration:

- If the interface has DNS server information configured or has obtained an AAA-authorized DNS server address, the device immediately sends an RA message containing the DNS server information.
- If the interface has no DNS server information specified or no AAA-authorized DNS server address assigned, no RA messages are triggered.

Each time the device sends an RA message from an interface, it immediately refreshes the RA message advertisement interval for that interface.

The same suppression mechanism applies when you enable or disable DNS suffix suppression in RA messages.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable DNS server suppression in RA messages.

   `ipv6 nd ra dns server suppress`

   By default, DNS server suppression in RA messages is disabled.

4. Enable DNS suffix suppression in RA messages.

   `ipv6 nd ra dns search-list suppress`

   By default, DNS suffix suppression in RA messages is disabled.

Setting the maximum number of attempts to send an NS message for DAD

About this task

An interface sends an NS message for DAD for an obtained IPv6 address. The interface resends the NS message if it does not receive a response within the time specified by the **`ipv6 nd ns retrans-timer`** command. If the interface receives no response after making the maximum attempts specified by the **`ipv6 nd dad attempts`** command, the interface uses the IPv6 address.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Set the number of attempts to send an NS message for DAD.

   `ipv6 nd dad attempts` *`times`*

The default setting is 1. When the *times* argument is set to 0, DAD is disabled.

Enabling ND proxy

About ND proxy

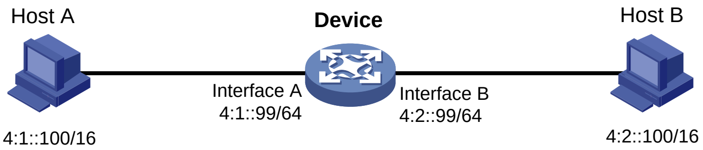
ND proxy enables a device to answer an NS message requesting the hardware address of a host on another network. With ND proxy, hosts in different broadcast domains can communicate with each other as they would on the same network.

ND proxy includes common ND proxy and local ND proxy.

Common ND proxy

As shown in Figure 6, Interface A with IPv6 address 4:1::99/64 and Interface B with IPv6 address 4:2::99/64 belong to different subnets. Host A and Host B reside on the same network but in different broadcast domains.

Figure 6 Application environment of ND proxy



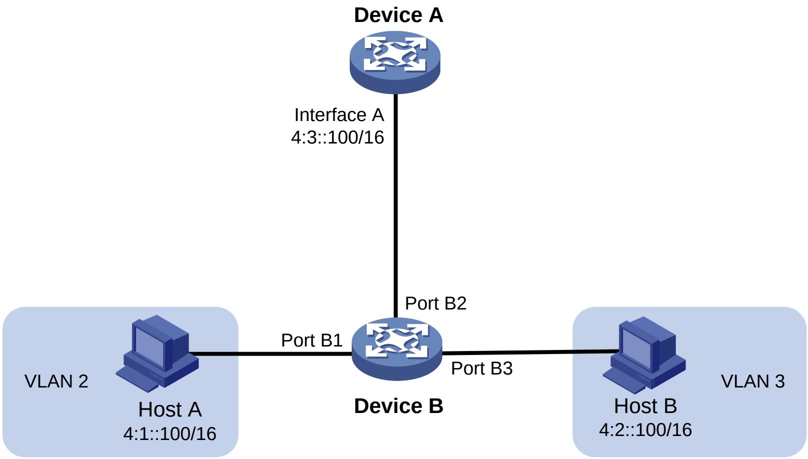
Because Host A's IPv6 address is on the same subnet as Host B's, Host A directly sends an NS message to obtain Host B's MAC address. However, Host B cannot receive the NS message because they belong to different broadcast domains.

To solve this problem, enable common ND proxy on Interface A and Interface B of the device. The device replies to the NS message from Host A, and forwards packets from other hosts to Host B.

Local ND proxy

As shown in Figure 7, Host A belongs to VLAN 2 and Host B belongs to VLAN 3. Host A and Host B connect to Port B1 and Port B3, respectively.

Figure 7 Application environment of local ND proxy



Because Host A's IPv6 address is on the same subnet as Host B's, Host A directly sends an NS message to obtain Host B's MAC address. However, Host B cannot receive the NS message because they belong to different VLANs.

To solve this problem, enable local ND proxy on Interface A of Device A so that Device A can forward messages between Host A and Host B.

Local ND proxy implements Layer 3 communication for two hosts in the following cases:

- The two hosts connect to ports of the same device and the ports must be in different VLANs.
- The two hosts connect to isolated Layer 2 ports in the same isolation group of a VLAN.

Enabling common ND proxy

1. Enter system view.
   `system-view`
2. Enter interface view.
   `interface` *`interface-type interface-number`*
3. Enable common ND proxy.
   `proxy-nd enable`
   By default, common ND proxy is disabled.

Enabling local ND proxy

1. Enter system view.
   `system-view`
2. Enter interface view.
   `interface` *`interface-type interface-number`*
3. Enable local ND proxy.
   `local-proxy-nd enable`
   By default, local ND proxy is disabled.

Display and maintenance commands for IPv6 ND

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the total number of neighbor entries.	**display ipv6 neighbors** { { **all** | **dynamic** | **static** } [**slot** *slot-number*] | **interface** *interface-type interface-number* | **vlan** *vlan-id* } **count**
Display neighbor information.	**display ipv6 neighbors** { { *ipv6-address* | **all** | **dynamic** | **static** } [**slot** *slot-number*] | **interface** *interface-type interface-number* | **vlan** *vlan-id* } [**verbose**]
Display neighbor information for a VPN.	**display ipv6 neighbors vpn-instance** *vpn-instance-name* [**count**]
Clear IPv6 neighbor information.	**reset ipv6 neighbors** { **all** | **dynamic** | **interface** *interface-type interface-number* | **slot** *slot-number* | **static** }

Contents

Configuring IPv6 fast forwarding

About IPv6 fast forwarding

Fast forwarding reduces route lookup time and improves packet forwarding efficiency by using a high-speed cache and data-flow-based technology. It identifies a data flow by using the following fields:

- Source IPv6 address.
- Destination IPv6 address.
- Source port number.
- Destination port number.
- Protocol number.

After a flow's first packet is forwarded through the routing table, fast forwarding creates an entry and uses the entry to forward subsequent packets of the flow.

Configuring IPv6 fast forwarding

About this task

The IPv6 fast forwarding feature will create fast forwarding entries for the device to speed up packet forwarding. When the traffic volume is high, the device will generate a large number of fast forwarding entries. These entries will cause high memory usage, which eventually leads to memory allocation failure for other services. In this case, you can disable this feature temporarily to free up device memory.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure IPv6 fast forwarding.
   - Enable IPv6 fast forwarding.

     ```
     ipv6 fast-forwarding enable
     ```

   - Disable IPv6 fast forwarding.

     ```
     undo ipv6 fast-forwarding enable
     ```

   By default, IPv6 fast forwarding is enabled.

Configuring the aging time for IPv6 fast forwarding entries

About this task

The IPv6 fast forwarding table uses an aging timer for each forwarding entry. If an entry is not updated before the timer expires, the device deletes the entry. If an entry has a hit within the aging time, the aging timer restarts.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the aging time for IPv6 fast forwarding entries.

   ```
   ipv6 fast-forwarding aging-time aging-time
   ```

   By default, the aging time is 30 seconds.

Configuring IPv6 fast forwarding load sharing

About this task

IPv6 fast forwarding load sharing enables the device to identify a data flow by using the packet information.

If IPv6 fast forwarding load sharing is disabled, the device identifies a data flow by the packet information and the input interface.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure IPv6 fast forwarding load sharing. Choose one option as needed:
   - Enable IPv6 fast forwarding load sharing.

     ```
     Ipv6 fast-forwarding load-sharing
     ```

   - Disable IPv6 fast forwarding load sharing.

     ```
     undo ipv6 fast-forwarding load-sharing
     ```

   By default, IPv6 fast forwarding load sharing is enabled.

Display and maintenance commands for IPv6 fast forwarding

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the aging time of the IPv6 fast forwarding entries.	**display ipv6 fast-forwarding aging-time**
Display IPv6 fast forwarding entries.	**display ipv6 fast-forwarding cache** [*ipv6-address*] [**slot** *slot-number*]
Display IPv6 fast forwarding entries for fragmented datagrams.	**display ipv6 fast-forwarding fragcache** [*ipv6-address*] [**slot** *slot-number*]
Clear the IPv6 fast forwarding table.	**reset ipv6 fast-forwarding cache** [**slot** *slot-number*]

Contents

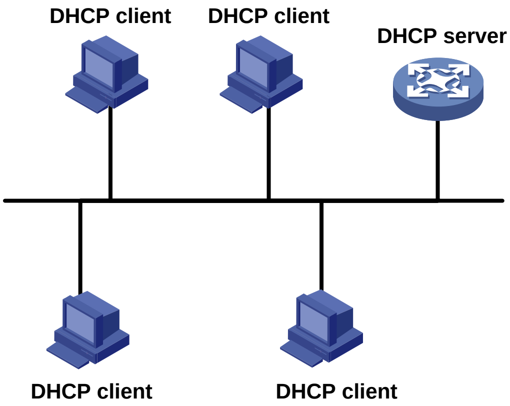
DHCP overview

DHCP network model

The Dynamic Host Configuration Protocol (DHCP) provides a framework to assign configuration information to network devices.

Figure 1 shows a typical DHCP application scenario where the DHCP clients and the DHCP server reside on the same subnet. The DHCP clients can also obtain configuration parameters from a DHCP server on another subnet through a DHCP relay agent. For more information about the DHCP relay agent, see "Configuring the DHCP relay agent."

Figure 1 A typical DHCP application



DHCP address allocation

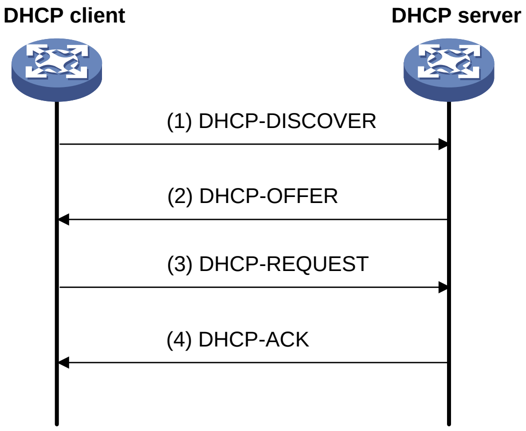
Allocation mechanisms

DHCP supports the following allocation mechanisms:

- **Static allocation**—The network administrator assigns an IP address to a client, such as a WWW server, and DHCP conveys the assigned address to the client.
- **Automatic allocation**—DHCP assigns a permanent IP address to a client.
- **Dynamic allocation**—DHCP assigns an IP address to a client for a limited period of time, which is called a lease. Most DHCP clients obtain their addresses in this way.

IP address allocation process

Figure 2 IP address allocation process



As shown in Figure 2, a DHCP server assigns an IP address to a DHCP client in the following process:

1. The client broadcasts a DHCP-DISCOVER message to locate a DHCP server.
2. Each DHCP server offers configuration parameters such as an IP address to the client in a DHCP-OFFER message. The sending mode of the DHCP-OFFER is determined by the flag field in the DHCP-DISCOVER message. For more information, see "DHCP message format."
3. If the client receives multiple offers, it accepts the first received offer, and broadcasts it in a DHCP-REQUEST message to formally request the IP address. (IP addresses offered by other DHCP servers can be assigned to other clients.)
4. All DHCP servers receive the DHCP-REQUEST message. However, only the server selected by the client does one of the following operations:
    - Returns a DHCP-ACK message to confirm that the IP address has been allocated to the client.
    - Returns a DHCP-NAK message to deny the IP address allocation.

After receiving the DHCP-ACK message, the client verifies the following details before using the assigned IP address:

- The assigned IP address is not in use. To verify this, the client broadcasts a gratuitous ARP packet. The assigned IP address is not in use if no response is received within the specified time.
- The assigned IP address is not on the same subnet as any IP address in use on the client.

Otherwise, the client sends a DHCP-DECLINE message to the server to request an IP address again.

IP address lease extension

A dynamically assigned IP address has a lease. When the lease expires, the IP address is reclaimed by the DHCP server. To continue using the IP address, the client must extend the lease duration.

When about half of the lease duration elapses, the DHCP client unicasts a DHCP-REQUEST to the DHCP server to extend the lease. Depending on the availability of the IP address, the DHCP server returns one of the following messages:

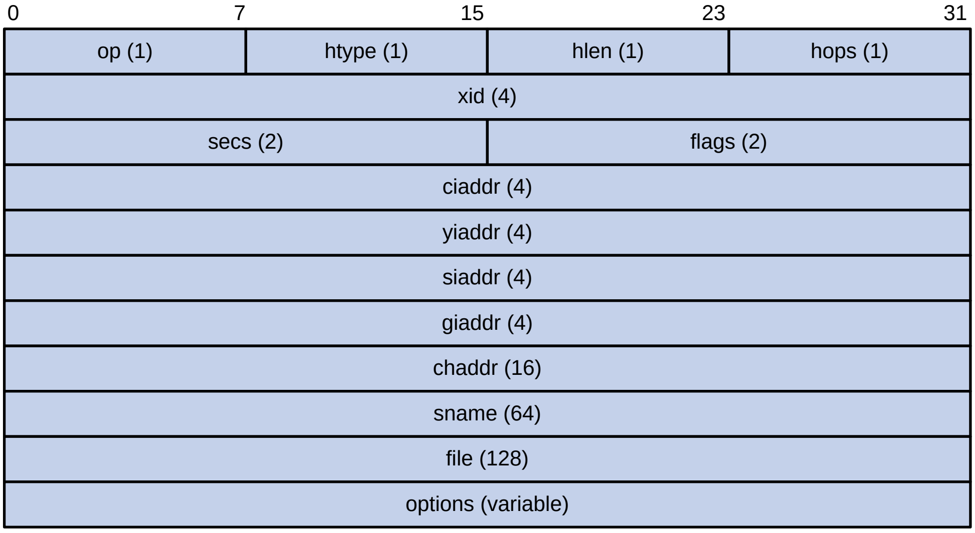
- A DHCP-ACK unicast confirming that the client's lease duration has been extended.
- A DHCP-NAK unicast denying the request.

If the client receives no reply, it broadcasts another DHCP-REQUEST message for lease extension when about seven-eighths of the lease duration elapses. Again, depending on the availability of the IP address, the DHCP server returns either a DHCP-ACK unicast or a DHCP-NAK unicast.

DHCP message format

Figure 3 shows the DHCP message format. DHCP uses some of the fields in significantly different ways. The numbers in parentheses indicate the size of each field in bytes.

Figure 3 DHCP message format

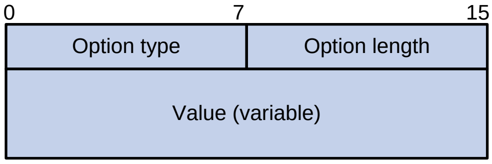


- **op**—Message type defined in options field. 1 = REQUEST, 2 = REPLY
- **htype**, **hlen**—Hardware address type and length of the DHCP client.
- **hops**—Number of relay agents a request message traveled.
- **xid**—Transaction ID, a random number chosen by the client to identify an IP address allocation.
- **secs**—Filled in by the client, the number of seconds elapsed since the client began address acquisition or renewal process. This field is reserved and set to 0.
- **flags**—The leftmost bit is defined as the BROADCAST (B) flag. If this flag is set to 0, the DHCP server sent a reply back by unicast. If this flag is set to 1, the DHCP server sent a reply back by broadcast. The remaining bits of the flags field are reserved for future use.
- **ciaddr**—Client IP address if the client has an IP address that is valid and usable. Otherwise, set to zero. (The client does not use this field to request an IP address to lease.)
- **yiaddr**—Your IP address. It is an IP address assigned by the DHCP server to the DHCP client.
- **siaddr**—Server IP address, from which the client obtained configuration parameters.
- **giaddr**—Gateway IP address. It is the IP address of the first relay agent to which a request message travels.
- **chaddr**—Client hardware address.
- **sname**—Server host name, from which the client obtained configuration parameters.
- **file**—Boot file (also called system software image) name and path information, defined by the server to the client.
- **options**—Optional parameters field that is variable in length. Optional parameters include the message type, lease duration, subnet mask, domain name server IP address, and WINS IP address.

DHCP options

DHCP extends the message format as an extension to BOOTP for compatibility. DHCP uses the options field to carry information for dynamic address allocation and provide additional configuration information for clients.

Figure 4 DHCP option format



Common DHCP options

The following are common DHCP options:

- **Option 3**—Router option. It specifies the gateway address to be assigned to the clients.
- **Option 6**—DNS server option. It specifies the DNS server IP address to be assigned to the clients.
- **Option 33**—Static route option. It specifies a list of classful static routes (the destination addresses in these static routes are classful) that a client should add into its routing table. If both Option 33 and Option 121 exist, Option 33 is ignored.
- **Option 51**—IP address lease option.
- **Option 53**—DHCP message type option. It identifies the type of the DHCP message.
- **Option 55**—Parameter request list option. It is used by a DHCP client to request specified configuration parameters. The option includes values that correspond to the parameters requested by the client.
- **Option 60**—Vendor class identifier option. A DHCP client uses this option to identify its vendor. A DHCP server uses this option to distinguish DHCP clients, and assigns IP addresses to them.
- **Option 66**—TFTP server name option. It specifies the TFTP server domain name to be assigned to the clients.
- **Option 67**—Boot file name option. It specifies the boot file name to be assigned to the client.
- **Option 121**—Classless route option. It specifies a list of classless static routes (the destination addresses in these static routes are classless) that a client should add into its routing table. If both Option 33 and Option 121 exist, Option 33 is ignored.
- **Option 150**—TFTP server IP address option. It specifies the TFTP server IP address to be assigned to the clients.

For more information about DHCP options, see RFC 2132 and RFC 3442.

Custom DHCP options

Some options, such as Option 43, Option 82, and Option 184, have no standard definitions in RFC 2132.

Vendor-specific option (Option 43)

Option 43 function

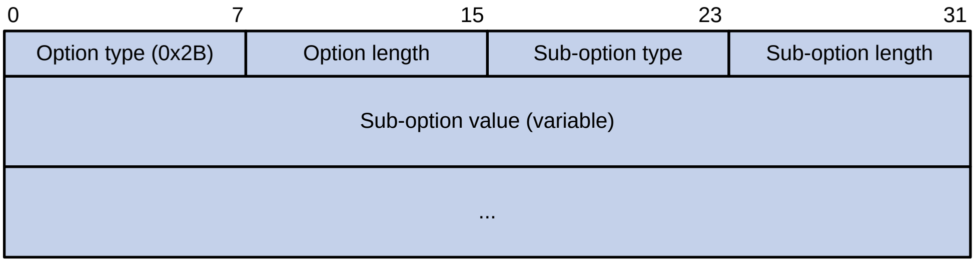
DHCP servers and clients use Option 43 to exchange vendor-specific configuration information.

The DHCP client can obtain the following information through Option 43:

- ACS parameters, including the ACS URL, username, and password.
- PXE server address, which is used to obtain the boot file or other control information from the PXE server.

Option 43 format

Figure 5 Option 43 format



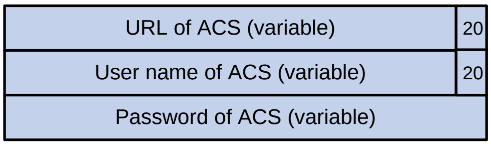
Network configuration parameters are carried in different sub-options of Option 43 as shown in Figure 5.

- **Sub-option type**—The field value can be 0x01 (ACS parameter sub-option), 0x02 (service provider identifier sub-option), or 0x80 (PXE server address sub-option).
- **Sub-option length**—Excludes the sub-option type and sub-option length fields.
- **Sub-option value**—The value format varies by sub-option.

Sub-option value field format

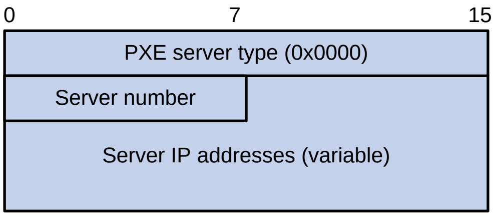
- **ACS parameter sub-option value field**—Includes the ACS URL, username, and password separated by spaces (hexadecimal number 20) as shown in Figure 6.

  Figure 6 ACS parameter sub-option value field



- **Service provider identifier sub-option value field**—Includes the service provider identifier.
- **PXE server address sub-option value field**—Includes the PXE server type that can only be 0, the server number that indicates the number of PXE servers contained in the sub-option, and server IP addresses, as shown in Figure 7.

  Figure 7 PXE server address sub-option value field



Relay agent option (Option 82)

Option 82 is the relay agent option. It records the location information about the DHCP client. When a DHCP relay agent or DHCP snooping device receives a client's request, it adds Option 82 to the request and sends it to the server.

The administrator can use Option 82 to locate the DHCP client and further implement security control and accounting. The DHCP server can use Option 82 to provide individual configuration policies for the clients.

Option 82 can include a maximum of 255 sub-options and must include a minimum of one sub-option. Option 82 supports the following sub-options: sub-option 1 (Circuit ID), and sub-option 2 (Remote ID). Option 82 has no standard definition. Its padding formats vary by vendor.

- Circuit ID has the following padding modes:
    - **String padding mode**—Includes a character string specified by the user.
    - **Normal padding mode**—Includes the VLAN ID and interface number of the interface that receives the client's request.
    - **Verbose padding mode**—Includes the access node identifier specified by the user, and the VLAN ID, interface number and interface type of the interface that receives the client's request.
- Remote ID has the following padding modes:
    - **String padding mode**—Includes a character string specified by the user.
    - **Normal padding mode**—Includes the MAC address of the DHCP relay agent interface or the MAC address of the DHCP snooping device that receives the client's request.
    - **Sysname padding mode**—Includes the name of the device. To set the device name, use the **`sysname`** command in system view.

Option 184

Option 184 is a reserved option. You can define the parameters in the option as needed. The device supports Option 184 carrying voice related parameters, so a DHCP client with voice functions can get voice parameters from the DHCP server.

Option 184 has the following sub-options:

- **Sub-option 1**—Specifies the IP address of the primary network calling processor. The primary processor acts as the network calling control source and provides program download services. For Option 184, you must define sub-option 1 to make other sub-options take effect.
- **Sub-option 2**—Specifies the IP address of the backup network calling processor. DHCP clients contact the backup processor when the primary one is unreachable.
- **Sub-option 3**—Specifies the voice VLAN ID and the result whether the DHCP client takes this VLAN as the voice VLAN.
- **Sub-option 4**—Specifies the failover route that includes the IP address and the number of the target user. A SIP VoIP user uses this IP address and number to directly establish a connection to the target SIP user when both the primary and backup calling processors are unreachable.

Protocols and standards

- RFC 2131, *Dynamic Host Configuration Protocol*
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
- RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*
- RFC 3046, *DHCP Relay Agent Information Option*

- RFC 3442, *The Classless Static Route Option for Dynamic Host Configuration Protocol (DHCP) version 4*

Configuring the DHCP server

About DHCP server

A DHCP server manages a pool of IP addresses and client configuration parameters. It selects an IP address and configuration parameters from the address pool and allocates them to a requesting DHCP client.

DHCP address assignment mechanisms

Configure the following address assignment mechanisms as needed:

- **Static address allocation**—Manually bind the MAC address or ID of a client to an IP address in a DHCP address pool. When the client requests an IP address, the DHCP server assigns the IP address in the static binding to the client.
- **Dynamic address allocation**—Specify IP address ranges in a DHCP address pool. Upon receiving a DHCP request, the DHCP server dynamically selects an IP address from the matching IP address range in the address pool.

You can specify IP address ranges in an address pool by using either of the following methods:

- **Method 1**—A primary subnet being divided into multiple address ranges in an address pool
- **Method 2**—A primary subnet and multiple secondary subnets in an address pool

A primary subnet being divided into multiple address ranges in an address pool

An address range includes a common IP address range and IP address ranges for DHCP user classes.

Upon receiving a DHCP request, the DHCP server finds a user class matching the client and selects an IP address in the address range of the user class for the client. A user class can include multiple matching rules, and a client matches the user class as long as it matches any of the rules. In address pool view, you can specify different address ranges for different user classes.

The DHCP server selects an IP address for a client by performing the following steps:

1. DHCP server compares the client against DHCP user classes in the order they are configured.
2. If the client matches a user class, the DHCP server selects an IP address from the address range of the user class.
3. If the matching user class has no assignable addresses, the DHCP server compares the client against the next user class. If all the matching user classes have no assignable addresses, the DHCP server selects an IP address from the common address range.
4. If the DHCP client does not match any DHCP user class, the DHCP server selects an address in the IP address range specified by the **`address range`** command. If the address range has no assignable IP addresses or it is not configured, the address allocation fails.

NOTE:

All address ranges must belong to the primary subnet. If an address range does not reside on the primary subnet, DHCP cannot assign the addresses in the address range.

A primary subnet and multiple secondary subnets in an address pool

The DHCP server selects an IP address from the primary subnet first. If there is no assignable IP address on the primary subnet, the DHCP server selects an IP address from secondary subnets in the order they are configured.

Principles for selecting an address pool

The DHCP server observes the following principles to select an address pool for a client:

1. If there is an address pool where an IP address is statically bound to the MAC address or ID of the client, the DHCP server selects this address pool and assigns the statically bound IP address and other configuration parameters to the client.
2. If the receiving interface has a DHCP policy and the DHCP client matches a user class, the DHCP server selects the address pool that is bound to the matching user class. If no matching user class is found, the server assigns an IP address and other parameters from the default DHCP address pool. If no default address pool is specified or the default address pool does not have assignable IP addresses, the address assignment fails.
3. If the receiving interface has an address pool applied, the DHCP server selects an IP address and other configuration parameters from this address pool.
4. If the above conditions are not met, the DHCP server selects an address pool depending on the client location.
   - **Client on the same subnet as the server**—The DHCP server compares the IP address of the receiving interface with the primary subnets of all address pools.
     - If a match is found, the server selects the address pool with the longest-matching primary subnet.
     - If no match is found, the DHCP server compares the IP address with the secondary subnets of all address pools. The server selects the address pool with the longest-matching secondary subnet.
   - **Client on a different subnet than the server**—The DHCP server compares the IP address in the **giaddr** field of the DHCP request with the primary subnets of all address pools.
     - If a match is found, the server selects the address pool with the longest-matching primary subnet.
     - If no match is found, the DHCP server compares the IP address with the secondary subnets of all address pools. The server selects the address pool with the longest-matching secondary subnet.

For example, two address pools 1.1.1.0/24 and 1.1.1.0/25 are configured but not applied to any DHCP server's interfaces.

- If the IP address of the receiving interface is 1.1.1.1/25, the DHCP server selects the address pool 1.1.1.0/25. If the address pool has no available IP addresses, the DHCP server will not select the other pool and the address allocation will fail.
- If the IP address of the receiving interface is 1.1.1.130/25, the DHCP server selects the address pool 1.1.1.0/24.

To ensure correct address allocation, keep the IP addresses used for dynamic allocation on one of the subnets:

- **Clients on the same subnet as the server**—Subnet where the DHCP server receiving interface resides.
- **Clients on a different subnet than the server**—Subnet where the first DHCP relay interface that faces the clients resides.

NOTE:

As a best practice, configure a minimum of one matching primary subnet in your network. Otherwise, the DHCP server selects only the first matching secondary subnet for address allocation. If the network has more DHCP clients than the assignable IP addresses in the secondary subnet, not all DHCP clients can obtain IP addresses.

IP address allocation sequence

The DHCP server selects an IP address for a client in the following sequence:

1. IP address statically bound to the client's MAC address or ID.
2. IP address that was ever assigned to the client.
3. IP address designated by the Option 50 field in the DHCP-DISCOVER message sent by the client.

   Option 50 is the Requested IP Address option. The client uses this option to specify the wanted IP address in a DHCP-DISCOVER message. The content of Option 50 is user defined.
4. First assignable IP address found in the way discussed in "DHCP address assignment mechanisms" and "Principles for selecting an address pool."
5. IP address that was a conflict or passed its lease duration. If no IP address is assignable, the server does not respond.

NOTE:

- If a client moves to another subnet, the DHCP server selects an IP address in the address pool matching the new subnet. It does not assign the IP address that was once assigned to the client.
- Conflicted IP addresses can be assigned to other DHCP clients only after the addresses are in conflict for an hour.

DHCP server tasks at a glance

To configure the DHCP server, perform the following tasks:

1. (Optional.) Creating a DHCP user class
2. Configuring an address pool on the DHCP server
3. (Optional.) Modifying the address pool selection method on the DHCP server
   - Applying an address pool to an interface
   - Configuring a DHCP policy for dynamic assignment
4. Enabling DHCP
5. Enabling the DHCP server on an interface
6. (Optional.) Configuring advanced DHCP features
   - Configuring IP address conflict detection
   - Enabling handling of Option 82
   - Configuring DHCP server compatibility
   - Setting the DSCP value for DHCP packets sent by the DHCP server
   - Configuring DHCP binding auto backup
   - Enabling client offline detection on the DHCP server
7. (Optional.) Configuring SNMP notification and logging
   - Configuring address pool usage alarming
   - Enabling DHCP logging on the DHCP server

Creating a DHCP user class

About this task

The DHCP server classifies DHCP users into different user classes according to the hardware address, option information, or the **giaddr** field in the received DHCP requests. The server allocates IP addresses and configuration parameters to DHCP clients in different user classes.

Procedure

1. Enter system view.

   `system-view`

2. Create a DHCP user class and enter DHCP user class view.

   `dhcp class` *`class-name`*

3. Configure a match rule for the DHCP user class.

   `if-match rule` *`rule-number`* { **`hardware-address`** *`hardware-address`* **`mask`** *`hardware-address-mask`* | **`option`** *`option-code`* [**`ascii`** *`ascii-string`* [**`offset`** *`offset`* | **`partial`**] | **`hex`** *`hex-string`* [**`mask`** *`mask`* | **`offset`** *`offset`* **`length`** *`length`* | **`partial`**]] | **`relay-agent`** *`gateway-address`* }

   By default, no match rule is configured for a DHCP user class.

Configuring an address pool on the DHCP server

DHCP address pool tasks at a glance

To configure a DHCP address pool, perform the following tasks:

1. Creating a DHCP address pool
2. Specifying IP address ranges in a DHCP address pool

   In one DHCP address pool, the two dynamic allocation methods cannot be both configured, but static and dynamic address allocations can be both implemented.

   - Specifying a primary subnet and multiple address ranges in a DHCP address pool
   - Specifying a primary subnet and multiple secondary subnets in a DHCP address pool
   - Configuring a static binding in a DHCP address pool
3. Specifying other configuration parameters to be assigned to DHCP clients
   - Specifying gateways for DHCP clients
   - Specifying a domain name suffix for DHCP clients
   - Specifying DNS servers for DHCP clients
   - Specifying WINS servers and NetBIOS node type for DHCP clients
   - Specifying BIMS server for DHCP clients
   - Specifying the configuration file for DHCP client automatic configuration
   - Specifying a server for DHCP clients
   - Configuring Option 184 parameters for DHCP clients
   - Customizing DHCP options
4. (Optional.) Applying a DHCP address pool to a VPN instance
5. (Optional.) Configuring the DHCP user class whitelist
6. (Optional.) Binding gateways to DHCP server's MAC address
7. (Optional.) Advertising subnets that are assigned to clients

Creating a DHCP address pool

1. Enter system view.

   system-view

2. Create a DHCP address pool and enter its view.

   dhcp server ip-pool *pool-name*

Specifying a primary subnet and multiple address ranges in a DHCP address pool

About this task

Some scenarios need to classify DHCP clients on the same subnet into different address groups. To meet this need, you can configure DHCP user classes and specify different address ranges for the classes. The clients matching a user class can then get the IP addresses of an address range. In addition, you can specify a common address range for the clients that do not match any user class. If no common address range is specified, such clients fail to obtain IP addresses.

If there is no need to classify clients, you do not need to configure DHCP user classes or their address ranges.

Restrictions and guidelines

- If you execute the **network** or **address range** command multiple times for the same address pool, the most recent configuration takes effect.
- If you execute the **forbidden-ip** or **forbidden-ip-range** command multiple times, you exclude multiple addresses from dynamic allocation.
- IP addresses specified by the **forbidden-ip** or **forbidden-ip-range** command are not assignable in the current address pool, but are assignable in other address pools. IP addresses specified by the **dhcp server forbidden-ip** command are not assignable in any address pool.
- You can use **class range** to modify an existing address range, and the new address range can include IP addresses that are being used by clients. Upon receiving a lease extension request for such an IP address, the DHCP server allocates a new IP address to the requesting client. But the original lease continues aging in the address pool, and will be released when the lease duration is reached. To release such lease without waiting for its timeout, execute the **reset dhcp server ip-in-use** command.

Procedure

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Specify the primary subnet in the address pool.

   network *network-address* [*mask-length* | **mask** *mask*]

   By default, no primary subnet is specified.

4. (Optional.) Specify the common address range.

   address range *start-ip-address end-ip-address*

   By default, no IP address range is specified.

5. (Optional.) Specify an IP address range for a DHCP user class.

   class *class-name* **range** *start-ip-address end-ip-address*

By default, no IP address range is specified for a user class.

The DHCP user class must already be created by using the **dhcp class** command.

6. (Optional.) Set the address lease duration.

   expired { **day** *day* [**hour** *hour* [**minute** *minute* [**second** *second*]]] | **unlimited** }

   The default setting is 1 day.

7. (Optional.) Exclude the specified IP addresses in the address pool from dynamic allocation.

   forbidden-ip *ip-address*&<1-8>

   By default, no IP addresses in the DHCP address pool are excluded from dynamic allocation.

8. (Optional.) Exclude the specified IP address range in the address pool from dynamic allocation.

   forbidden-ip-range *start-ip-address* [*end-ip-address*]

   By default, no IP address ranges in the DHCP address pool are excluded from dynamic allocation.

9. (Optional.) Exclude the specified IP addresses from automatic allocation in system view.

   a. Return to system view.

      quit

   b. Exclude the specified IP addresses from automatic allocation globally.

      dhcp server forbidden-ip *start-ip-address* [*end-ip-address*] [**vpn-instance** *vpn-instance-name*]

      By default, except for the IP address of the DHCP server interface, IP addresses in all address pools are assignable.

Specifying a primary subnet and multiple secondary subnets in a DHCP address pool

About this task

If an address pool has a primary subnet and multiple secondary subnets, the server assigns IP addresses on a secondary subnet when the primary subnet has no assignable IP addresses.

Restrictions and guidelines

IP addresses specified by the **forbidden-ip** or **forbidden-ip-range** command are not assignable in the current address pool, but are assignable in other address pools. IP addresses specified by the **dhcp server forbidden-ip** command are not assignable in any address pool.

Specifying a primary subnet and multiple secondary subnets

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Specify the primary subnet.

   network *network-address* [*mask-length* | **mask** *mask*]

   By default, no primary subnet is specified.

   You can specify only one primary subnet in each address pool. If you execute the **network** command multiple times, the most recent configuration takes effect.

4. (Optional.) Specify a secondary subnet.

   network *network-address* [*mask-length* | **mask** *mask*] **secondary**

   By default, no secondary subnet is specified.

You can specify a maximum of 32 secondary subnets in one address pool.

5. (Optional.) Return to address pool view.

   quit

Setting the lease duration for dynamically allocation IP addresses

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Set the address lease duration.

   expired { **day** *day* [**hour** *hour* [**minute** *minute* [**second** *second*]]] | **unlimited** }

   The default setting is 1 day.

Excluding IP addresses from dynamic allocation

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Exclude the specified IP addresses from dynamic allocation.

   forbidden-ip *ip-address*&<1-8>

   By default, no IP addresses in the DHCP address pool are excluded from dynamic allocation.

   To exclude multiple addresses from the address pool, repeat this step.

4. (Optional.) Exclude the specified IP address range in the address pool from dynamic allocation.

   forbidden-ip-range *start-ip-address* [*end-ip-address*]

   By default, no IP address ranges in the DHCP address pool are excluded from dynamic allocation.

   To exclude multiple address ranges from the address pool, repeat this step.

5. (Optional.) Exclude the specified IP addresses from dynamic allocation in system view.

   a. Return to system view.

      quit

   b. Exclude the specified IP addresses from dynamic allocation globally.

      dhcp server forbidden-ip *start-ip-address* [*end-ip-address*] [**vpn-instance** *vpn-instance-name*]

      By default, except for the IP address of the DHCP server interface, IP addresses in all address pools are assignable.

      To exclude multiple address ranges globally, repeat this step.

Configuring a static binding in a DHCP address pool

About this task

Some DHCP clients, such as a WWW server, need fixed IP addresses. To provide a fixed IP address for a client, you can statically bind the MAC address or ID of the client to an IP address in a DHCP address pool. When the client requests an IP address, the DHCP server assigns the IP address in the static binding to the client.

Restrictions and guidelines

- The IP address of a static binding cannot be the address of the DHCP server interface. Otherwise, an IP address conflict occurs and the bound client cannot obtain an IP address correctly.
- Multiple interfaces on the same device might all use DHCP to request a static IP address. In this case, use client IDs rather than the device's MAC address to identify the interfaces. Otherwise, IP address allocation will fail.

Procedure

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

3. Configure a static binding.

   `static-bind ip-address` *`ip-address`* [*`mask-length`* | **`mask`** *`mask`*] { **`client-identifier`** *`client-identifier`* | **`hardware-address`** *`hardware-address`* [**`ethernet`** | **`token-ring`**] } [**`description`** *`description-text`*]

   By default, no static binding is configured.

   One IP address can be bound to only one client MAC or client ID. You cannot modify bindings that have been created. To change the binding for a DHCP client, you must delete the existing binding first.

4. (Optional.) Set the lease duration for the IP address.

   `expired` { **`day`** *`day`* [**`hour`** *`hour`* [**`minute`** *`minute`* [**`second`** *`second`*]]] | **`unlimited`** }

   By default, the lease duration is 1 day.

Specifying gateways for DHCP clients

About this task

DHCP clients send packets destined for other networks to a gateway. The DHCP server can assign the gateway address to the DHCP clients.

Restrictions and guidelines

You can specify gateway addresses in each address pool on the DHCP server. A maximum of 64 gateways can be specified in DHCP address pool view or secondary subnet view.

The DHCP server assigns gateway addresses to clients on a secondary subnet in the following ways:

- If gateways are specified in both address pool view and secondary subnet view, DHCP assigns those specified in the secondary subnet view.
- If gateways are specified in address pool view but not in secondary subnet view, DHCP assigns those specified in address pool view.

Procedure

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

3. Specify gateways.

gateway-list *ip-address*&<1-64>

By default, no gateway is specified.

4. (Optional.) Specify gateways in secondary subnet view.
   a. Enter secondary subnet view.
      network *network-address* [*mask-length* | **mask** *mask*] **secondary**
   b. Specify gateways.
      gateway-list *ip-address*&<1-64>
      By default, no gateway is specified.

Specifying a domain name suffix for DHCP clients

About this task

You can specify a domain name suffix in a DHCP address pool on the DHCP server. With this suffix assigned, the client only needs to input part of a domain name, and the system adds the domain name suffix for name resolution. For more information about DNS, see "Configuring DNS."

Procedure

1. Enter system view.
   system-view
2. Enter DHCP address pool view.
   dhcp server ip-pool *pool-name*
3. Specify a domain name suffix.
   domain-name *domain-name*
   By default, no domain name is specified.

Specifying DNS servers for DHCP clients

About this task

To access hosts on the Internet through domain names, a DHCP client must contact a DNS server to resolve names. You can specify up to eight DNS servers in a DHCP address pool.

Procedure

1. Enter system view.
   system-view
2. Enter DHCP address pool view.
   dhcp server ip-pool *pool-name*
3. Specify DNS servers.
   dns-list *ip-address*&<1-8>
   By default, no DNS server is specified.

Specifying WINS servers and NetBIOS node type for DHCP clients

About this task

A Microsoft DHCP client using NetBIOS protocol must contact a WINS server for name resolution.

In addition, you must specify one of the following NetBIOS node types to approach name resolution:

- **b (broadcast)-node**—A b-node client sends the destination name in a broadcast message. The destination returns its IP address to the client after receiving the message.
- **p (peer-to-peer)-node**—A p-node client sends the destination name in a unicast message to the WINS server. The WINS server returns the destination IP address.
- **m (mixed)-node**—An m-node client broadcasts the destination name. If it receives no response, it unicasts the destination name to the WINS server to get the destination IP address.
- **h (hybrid)-node**—An h-node client unicasts the destination name to the WINS server. If it receives no response, it broadcasts the destination name to get the destination IP address.

Procedure

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

   By default, no DHCP address pool exists.

3. Specify WINS servers.

   nbns-list *ip-address*&<1-8>

   By default, no WINS server is specified.

   This step is optional for b-node. You can specify a maximum of eight WINS servers for such clients in one DHCP address pool.

4. Specify the NetBIOS node type.

   netbios-type { **b-node** | **h-node** | **m-node** | **p-node** }

   By default, no NetBIOS node type is specified.

Specifying BIMS server for DHCP clients

About this task

Perform this task to provide the BIMS server IP address, port number, and shared key for the clients. The DHCP clients contact the BIMS server to get configuration files and perform software upgrade and backup.

Procedure

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Specify the BIMS server IP address, port number, and shared key.

   bims-server ip *ip-address* [**port** *port-number*] **sharekey** { **cipher** | **simple** } *string*

   By default, no BIMS server information is specified.

Specifying the configuration file for DHCP client automatic configuration

About this task

Automatic configuration enables a device to automatically obtain a set of configuration settings at startup. The server-based automatic configuration requires the cooperation of the DHCP server and file server (TFTP or HTTP server). The device uses the obtained parameters to contact the file server to get the configuration file. For more information about automatic configuration, see *Fundamentals Configuration Guide*.

Specifying the configuration file on a TFTP file server

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

   By default, no DHCP address pool exists.

3. Specify the IP address or the name of a TFTP server.
   - Specify the IP address of the TFTP server.

     `tftp-server ip-address` *`ip-address`*

     By default, no TFTP server IP address is specified.
   - Specify the name of the TFTP server.

     `tftp-server domain-name` *`domain-name`*

     By default, no TFTP server name is specified.

4. Specify the configuration file name.

   `bootfile-name` *`bootfile-name`*

   By default, no configuration file name is specified.

Specifying the URL of the configuration file on an HTTP file server

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

3. Specify the URL of the configuration file.

   `bootfile-name` *`url`*

   By default, no configuration file URL is specified.

Specifying a server for DHCP clients

About this task

Some DHCP clients need to obtain configuration information from a server, such as a TFTP server. You can specify the IP address of that server. The DHCP server sends the server's IP address to DHCP clients along with other configuration information.

Procedure

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

3. Specify the IP address of a server.

   `next-server` *`ip-address`*

   By default, no server is specified.

Configuring Option 184 parameters for DHCP clients

About this task

To assign calling parameters to DHCP clients with voice service, you must configure Option 184 on the DHCP server. For more information about Option 184, see "Option 184."

Procedure

1. Enter system view.
   system-view
2. Enter DHCP address pool view.
   dhcp server ip-pool *pool-name*
3. Specify the IP address of the primary network calling processor.
   voice-config ncp-ip *ip-address*
   By default, no primary network calling processor is specified.
   After you configure this command, the other Option 184 parameters take effect.
4. (Optional.) Specify the IP address of the backup server.
   voice-config as-ip *ip-address*
   By default, no backup network calling processor is specified.
5. (Optional.) Configure the voice VLAN.
   voice-config voice-vlan *vlan-id* { **disable** | **enable** }
   By default, no voice VLAN is configured.
6. (Optional.) Specify the failover IP address and dialer string.
   voice-config fail-over *ip-address dialer-string*
   By default, no failover IP address or dialer string is specified.

Customizing DHCP options

DHCP option customization applications

You can customize DHCP options for the following purposes:

- Add newly released options.
- Add options for which the vendor defines the contents, for example, Option 43.
- Add options for which the CLI does not provide a dedicated configuration command. For example, you can use the **option 4 ip-address 1.1.1.1** command to define the time server address 1.1.1.1 for DHCP clients.
- Add all option values if the actual requirement exceeds the limit for a dedicated option configuration command. For example, the **dns-list** command can specify up to eight DNS servers. To specify more than eight DNS servers, you must use the **option 6** command to define all DNS servers.

Common DHCP options

Table 1 lists common DHCP options and their parameters.

Table 1 Common DHCP options

Option	Option name	Corresponding command	Recommended parameter in the option command
3	Router Option	**gateway-list**	**ip-address**

Option	Option name	Corresponding command	Recommended parameter in the option command
6	Domain Name Server Option	**dns-list**	**ip-address**
15	Domain Name	**domain-name**	**ascii**
44	NetBIOS over TCP/IP Name Server Option	**nbns-list**	**ip-address**
46	NetBIOS over TCP/IP Node Type Option	**netbios-type**	**hex**
66	TFTP server name	**tftp-server**	**ascii**
67	Boot file name	**bootfile-name**	**ascii**
43	Vendor Specific Information	N/A	**hex**

Restrictions and guidelines

Use caution when customizing DHCP options because the configuration might affect DHCP operation.

You can customize a DHCP option in a DHCP address pool

You can customize a DHCP option in a DHCP option group, and specify the option group for a user class in an address pool. A DHCP client in the user class will obtain the option configuration.

Customizing a DHCP option in a DHCP address pool

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   dhcp server ip-pool *pool-name*

3. Customize a DHCP option.

   option *code* { **ascii** *ascii-string* | **hex** *hex-string* | **ip-address** *ip-address*&<1-8> }

   By default, no DHCP option is customized in a DHCP address pool.

   DHCP options specified in DHCP option groups take precedence over those specified in DHCP address pools.

Customizing a DHCP option in a DHCP option group

1. Enter system view.

   system-view

2. Create a DHCP option group and enter DHCP option group view.

   dhcp option-group *option-group-number*

3. Customize a DHCP option.

   option *code* { **ascii** *ascii-string* | **hex** *hex-string* | **ip-address** *ip-address*&<1-8> }

   By default, no DHCP option is customized in a DHCP option group.

   If multiple DHCP option groups have the same option, the server selects the option in the DHCP option group first matching the user class.

4. Return to system view.

   quit

5. Enter DHCP address pool view.

```
dhcp server ip-pool pool-name
```

6. Specify the DHCP option group for the DHCP user class.

   ```
   class class-name option-group option-group-number
   ```

   By default, no DHCP option group is specified for a DHCP user class.

Applying a DHCP address pool to a VPN instance

About this task

If a DHCP address pool is applied to a VPN instance, the DHCP server assigns IP addresses in this address pool to clients in the VPN instance. Addresses in this address pool will not be assigned to clients on the public network.

The DHCP server can obtain the VPN instance to which a DHCP client belongs from the following information:

- The client's VPN information stored in authentication modules, such as IPoE.
- The VPN information of the DHCP server's interface that receives DHCP packets from the client.

If both VPN instances can be obtained, the VPN information from authentication modules takes priority over the VPN information of the receiving interface.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DHCP address pool view.

   ```
   dhcp server ip-pool pool-name
   ```

3. Apply the address pool to a VPN instance.

   ```
   vpn-instance vpn-instance-name
   ```

   By default, the address pool is not applied to any VPN instance.

Configuring the DHCP user class whitelist

About this task

The DHCP user class whitelist allows the DHCP server to process requests only from clients on the DHCP user class whitelist.

Restrictions and guidelines

The whitelist does not take effect on clients who request static IP addresses, and the server always processes their requests.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DHCP address pool view.

   ```
   dhcp server ip-pool pool-name
   ```

3. Enable the DHCP user class whitelist.

   ```
   verify class
   ```

   By default, the DHCP user class whitelist is disabled.

4. Add DHCP user classes to the DHCP user class whitelist.

```
valid class class-name&<1-8>
```

By default, no DHCP user class is on the DHCP user class whitelist.

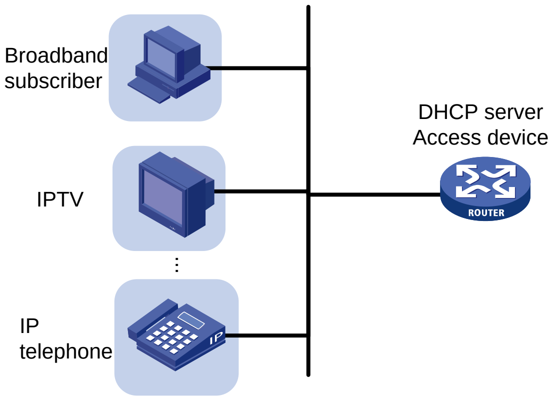
Binding gateways to DHCP server's MAC address

About this task

This feature enables the DHCP server to assign different gateway IP addresses to DHCP clients. In addition, the DHCP server uses the gateway IP addresses and the server's MAC address to reply to ARP requests from the clients.

As shown in Figure 8, the DHCP server is configured on the access device that provides access for clients of different service types, such as broadband, IPTV, and IP telephone. The clients of different types obtain IP addresses on different subnets. For the clients to access the network, the access interface typically has no IP address configured. You must bind the gateways to the DHCP server's MAC address when specifying gateways for the DHCP clients.

Figure 8 Network diagram



Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter DHCP address pool view.

   ```
   dhcp server ip-pool pool-name
   ```

3. Bind the gateways to the device's MAC address.

   ```
   gateway-list ip-address&<1-64> export-route
   ```

   By default, gateways are not bound to any MAC address.

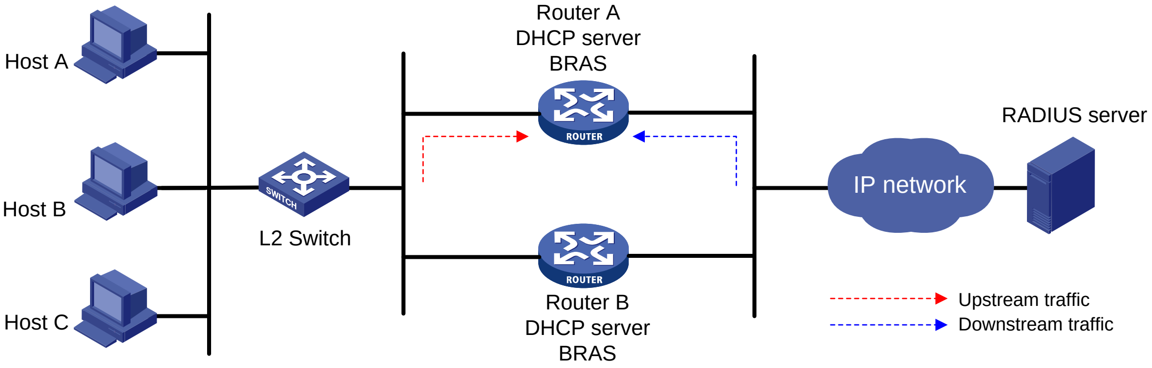
Advertising subnets that are assigned to clients

About this task

This feature enables the route management module to advertise subnets assigned to DHCP clients. This feature achieves symmetric routing for traffic of the same host.

As shown in Figure 9, Router A and Router B act as both the DHCP server and the BRAS device. The BRAS devices send accounting packets to the RADIUS server. To enable the BRAS devices to collect correct accounting information for each RADIUS user, configure the DHCP server to advertise subnets assigned to clients. The upstream and downstream traffic of a RADIUS user will pass through the same BRAS device.

Figure 9 Network diagram



Procedure

1. Enter system view.

   `system-view`

2. Create a DHCP address pool and enter its view.

   `dhcp server ip-pool` *`pool-name`*

3. Advertise subnets assigned to DHCP clients.

   `network` *`network-address`* [*`mask-length`* | **`mask`** *`mask`*] [**`secondary`**] **`export-route`**

   By default, the subnets assigned to DHCP clients are not advertised.

Applying an address pool to an interface

About this task

Perform this task to apply a DHCP address pool to an interface.

Upon receiving a DHCP request from the interface, the DHCP server performs address allocation in the following ways:

- If a static binding is found for the client, the server assigns the static IP address and configuration parameters from the address pool that contains the static binding.
- If no static binding is found for the client, the server uses the address pool applied to the interface for address and configuration parameter allocation.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Apply an address pool to the interface.

   `dhcp server apply ip-pool` *`pool-name`*

   By default, no address pool is applied to an interface.

   If the applied address pool does not exist, the DHCP server fails to perform dynamic address allocation.

Configuring a DHCP policy for dynamic assignment

About this task

In a DHCP policy, each DHCP user class has a bound DHCP address pool. Clients matching different user classes obtain IP addresses and other parameters from different address pools. The DHCP policy must be applied to the interface that acts as the DHCP server. When receiving a DHCP request, the DHCP server compares the packet against the user classes in the order that they are configured.

- If a matching user class is found and the bound address pool has assignable IP addresses, the server assigns an IP address and other parameters from the address pool. If the address pool does not have assignable IP addresses, the address assignment fails.
- If no match is found, the server assigns an IP address and other parameters from the default DHCP address pool. If no default address pool is specified or the default address pool does not have assignable IP addresses, the address assignment fails.

For successful address assignment, make sure the applied DHCP policy and the bound address pools exist.

Restrictions and guidelines

A DHCP policy take effect only after it is applied to an interface.

Procedure

1. Enter system view.

   `system-view`

2. Create a DHCP policy and enter DHCP policy view.

   `dhcp policy` *`policy-name`*

3. Specify a DHCP address pool for a DHCP user class.

   `class` *`class-name`* **`ip-pool`** *`pool-name`*

   By default, no address pool is specified for a user class.

4. Specify the default DHCP address pool.

   `default ip-pool` *`pool-name`*

   By default, no default address pool is specified.

5. Return to system view.

   `quit`

6. Enter interface view.

   `interface` *`interface-type interface-number`*

7. Apply the DHCP policy to the interface.

   `dhcp apply-policy` *`policy-name`*

   By default, no DHCP policy is applied to an interface.

Enabling DHCP

Restrictions and guideline

You must enable DHCP to make other DHCP configurations take effect.

Procedure

1. Enter system view.

   system-view

2. Enable DHCP.

   dhcp enable

   By default, DHCP is disabled.

Enabling the DHCP server on an interface

About this task

Perform this task to enable the DHCP server on an interface. Upon receiving a DHCP request on the interface, the DHCP server assigns the client an IP address and other configuration parameters from a DHCP address pool.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the DHCP server on the interface.

   dhcp select server

   By default, the DHCP server is enabled on the interface.

Configuring IP address conflict detection

About this task

Before assigning an IP address, the DHCP server pings that IP address.

- If the server receives a response within the specified period, it selects and pings another IP address.
- If it receives no response, the server continues to ping the IP address until the maximum number of ping packets are sent. If still no response is received, the server assigns the IP address to the requesting client. The DHCP client uses gratuitous ARP to perform IP address conflict detection.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the maximum number of ping packets to be sent for conflict detection.

   dhcp server ping packets *number*

   The default setting is one.

   To disable IP address conflict detection, set the value to **0**.

3. (Optional.) Set the ping timeout time.

   dhcp server ping timeout *milliseconds*

   The default setting is 500 ms.

   To disable IP address conflict detection, set the value to **0**.

Enabling handling of Option 82

About this task

Perform this task to enable the DHCP server to handle Option 82. Upon receiving a DHCP request that contains Option 82, the DHCP server adds Option 82 into the DHCP response.

If you disable the DHCP to handle Option 82, it does not add Option 82 into the response message.

You must enable handling of Option 82 on both the DHCP server and the DHCP relay agent to ensure correct processing for Option 82. For information about enabling handling of Option 82 on the DHCP relay agent, see "Configuring DHCP relay agent support for Option 82."

Procedure

1. Enter system view.

   `system-view`

2. Enable the server to handle Option 82.

   `dhcp server relay information enable`

   By default, handling of Option 82 is enabled.

Configuring DHCP server compatibility

Perform this task to enable the DHCP server to support DHCP clients that are incompliant with RFC.

Configuring the DHCP server to always broadcast responses

About this task

By default, the DHCP server broadcasts a response only when the broadcast flag in the DHCP request is set to 1. You can configure the DHCP server to ignore the broadcast flag and always broadcast a response. This feature is useful when some clients set the broadcast flag to 0 but do not accept unicast responses.

The DHCP server always unicasts a response in the following situations, regardless of whether this feature is configured or not:

- The DHCP request is from a DHCP client that has an IP address (the **ciaddr** field is not 0).
- The DHCP request is forwarded by a DHCP relay agent from a DHCP client (the **giaddr** field is not 0).

Procedure

1. Enter system view.

   `system-view`

2. Enable the DHCP server to always broadcast all responses.

   `dhcp server always-broadcast`

   By default, the DHCP server reads the broadcast flag to decide whether to broadcast or unicast a response.

Configuring the DHCP server to ignore BOOTP requests

About this task

The lease duration of the IP addresses obtained by the BOOTP clients is unlimited. For some scenarios that do not allow unlimited leases, you can configure the DHCP server to ignore BOOTP requests.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the DHCP server to ignore BOOTP requests.

   ```
   dhcp server bootp ignore
   ```

   By default, the DHCP server processes BOOTP requests.

Configuring the DHCP server to send BOOTP responses in RFC 1048 format

About this task

Not all BOOTP clients can send requests that are compatible with RFC 1048. By default, the DHCP server does not process the Vend field of RFC 1048-incompliant requests but copies the Vend field into responses.

This feature enables the DHCP server to fill the Vend field in RFC 1048-compliant format in DHCP responses to RFC 1048-incompliant requests sent by BOOTP clients.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the DHCP server to send BOOTP responses in RFC 1048 format to the RFC 1048-incompliant BOOTP requests.

   ```
   dhcp server bootp reply-rfc-1048
   ```

   By default, the DHCP server directly copies the Vend field of such requests into the responses.

Disabling Option 60 encapsulation in DHCP replies

About this task

If one or more DHCP clients cannot resolve Option 60, disable the DHCP server from encapsulating Option 60 in DHCP replies. If you do not disable the capability, the DHCP server encapsulates Option 60 in a DHCP reply in the following situations:

- The received DHCP packet contains Option 60.
- Option 60 is configured for the address pool.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Disable the DHCP server from encapsulating Option 60 in DHCP replies.

   ```
   dhcp server reply-exclude-option60
   ```

   By default, the DHCP server can encapsulate Option 60 in DHCP replies.

Setting the DSCP value for DHCP packets sent by the DHCP server

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.

   system-view

2. Set the DSCP value for DHCP packets sent by the DHCP server.

   dhcp dscp *dscp-value*

   By default, the DSCP value in DHCP packets sent by the DHCP server is 56.

Configuring DHCP binding auto backup

About this task

The auto backup feature saves bindings to a backup file and allows the DHCP server to download the bindings from the backup file at the server reboot. The bindings include the lease bindings and conflicted IP addresses. They cannot survive a reboot on the DHCP server.

The DHCP server does not provide services during the download process. If a connection error occurs during the process and cannot be repaired in a short amount of time, you can terminate the download operation. Manual interruption allows the DHCP server to provide services without waiting for the connection to be repaired.

Procedure

1. Enter system view.

   system-view

2. Configure the DHCP server to back up the bindings to a file.

   dhcp server database filename { *filename* | **url** *url* [**username** *username* [**password** { **cipher** | **simple** } *string*]] }

   By default, the DHCP server does not back up the DHCP bindings.

   With this command executed, the DHCP server backs up its bindings immediately and runs auto backup.

3. (Optional.) Manually save the DHCP bindings to the backup file.

   dhcp server database update now

4. (Optional.) Set the waiting time after a DHCP binding change for the DHCP server to update the backup file.

   dhcp server database update interval *interval*

   By default, the DHCP server waits 300 seconds to update the backup file after a DHCP binding change. If no DHCP binding changes, the backup file is not updated.

5. (Optional.) Terminate the download of DHCP bindings from the backup file.

   dhcp server database update stop

   This command only triggers one termination.

Enabling client offline detection on the DHCP server

About this task

The client offline detection feature reclaims an assigned IP address and deletes the binding entry when the ARP entry for the IP address ages out.

Restrictions and guidelines

The feature does not function if an ARP entry is manually deleted.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable client offline detection.

   `dhcp client-detect`

   By default, client offline detection is disabled on the DHCP server.

Configuring address pool usage alarming

About this task

Perform this task to set the threshold for address pool usage alarming. When the threshold is exceeded, the system sends notifications to the SNMP module. For DHCP notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter DHCP address pool view.

   `dhcp server ip-pool` *`pool-name`*

3. (Optional.) Set the threshold for address pool usage alarming.

   `ip-in-use threshold` *`threshold-value`*

   The default threshold is 100%.

Enabling DHCP logging on the DHCP server

About this task

The DHCP logging feature enables the DHCP server to generate DHCP logs and send them to the information center. The information helps administrators locate and solve problems. For information about the log destination and output rule configuration in the information center, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

As a best practice, disable this feature if the log generation affects the device performance or reduces the address allocation efficiency. For example, this situation might occur when a large number of clients frequently come online or go offline.

Procedure

1. Enter system view.
   system-view
2. Enable DHCP logging.
   dhcp log enable
   By default, DHCP logging is disabled.

Display and maintenance commands for DHCP server

IMPORTANT:

A restart of the DHCP server or execution of the **reset dhcp server ip-in-use** command deletes all lease information. The DHCP server denies any DHCP request for lease extension, and the client must request an IP address again.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about IP address conflicts.	**display dhcp server conflict** [**ip** *ip-address*] [**vpn-instance** *vpn-instance-name*]
Display information about DHCP binding auto backup.	**display dhcp server database**
Display information about lease-expired IP addresses.	**display dhcp server expired** [[**ip** *ip-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Display information about assignable IP addresses.	**display dhcp server free-ip** [**pool** *pool-name* | **vpn-instance** *vpn-instance-name*]
Display information about assigned IP addresses.	**display dhcp server ip-in-use** [[**ip** *ip-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Display information about DHCP address pools.	**display dhcp server pool** [*pool-name* | **vpn-instance** *vpn-instance-name*]
Display DHCP server statistics.	**display dhcp server statistics** [**pool** *pool-name* | **vpn-instance** *vpn-instance-name*]
Clear information about IP address conflicts.	**reset dhcp server conflict** [**ip** *ip-address*] [**vpn-instance** *vpn-instance-name*]
Clear information about lease-expired IP	**reset dhcp server expired** [[**ip**

Task	Command
addresses.	*ip-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Clear information about assigned IP addresses.	**reset dhcp server ip-in-use** [[**ip** *ip-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Clear DHCP server statistics.	**reset dhcp server statistics** [**vpn-instance** *vpn-instance-name*]

Troubleshooting DHCP server configuration

Failure to obtain a non-conflicting IP address

Symptom

A client's IP address obtained from the DHCP server conflicts with an IP address of another host.

Solution

Another host on the subnet might have the same IP address.

To resolve the problem:

1. Disable the client's network adapter or disconnect the client's network cable. Ping the IP address of the client from another host to check whether there is a host using the same IP address.
2. If a ping response is received, the IP address has been manually configured on a host. Execute the **dhcp server forbidden-ip** command on the DHCP server to exclude the IP address from dynamic allocation.
3. Enable the network adapter or connect the network cable, release the IP address, and obtain another one on the client. For example, to release the IP address and obtain another one on a Windows XP DHCP client:
   a. In Windows environment, execute the **cmd** command to enter the DOS environment.
   b. Enter **ipconfig /release** to relinquish the IP address.
   c. Enter **ipconfig /renew** to obtain another IP address.

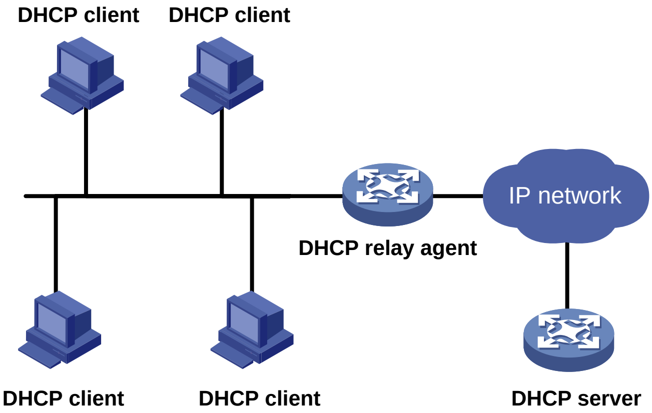
Configuring the DHCP relay agent

About DHCP relay agent

The DHCP relay agent enables clients to get IP addresses and configuration parameters from a DHCP server on another subnet.

Figure 10 shows a typical application of the DHCP relay agent.

Figure 10 DHCP relay agent application

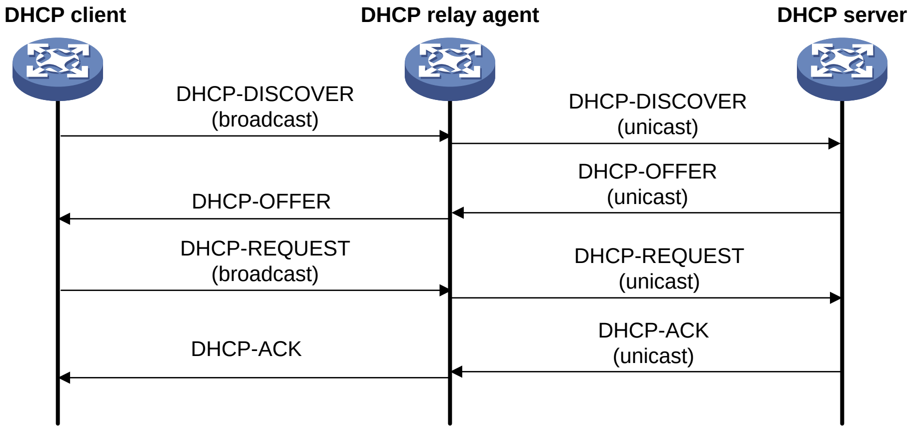


DHCP relay agent operation

The DHCP server and client interact with each other in the same way regardless of whether the relay agent exists. For the interaction details, see "IP address allocation process." The following only describes steps related to the DHCP relay agent:

1. After receiving a DHCP-DISCOVER or DHCP-REQUEST broadcast message from a DHCP client, the DHCP relay agent processes the message as follows:
   a. Fills the **giaddr** field of the message with its IP address.
   b. Unicasts the message to the designated DHCP server.
2. Based on the **giaddr** field, the DHCP server returns an IP address and other configuration parameters in a response.
3. The relay agent conveys the response to the client.

Figure 11 DHCP relay agent operation



DHCP relay agent support for Option 82

Option 82 records the location information about the DHCP client. It enables the administrator to perform the following tasks:

- Locate the DHCP client for security and accounting purposes.
- Assign IP addresses in a specific range to clients.

For more information about Option 82, see "Relay agent option (Option 82)."

If the DHCP relay agent supports Option 82, it handles DHCP requests by following the strategies described in Table 2.

If a response returned by the DHCP server contains Option 82, the DHCP relay agent removes the Option 82 before forwarding the response to the client.

Table 2 Handling strategies of the DHCP relay agent

If a DHCP request has…	Handling strategy	The DHCP relay agent…
Option 82	Drop	Drops the message.
	Keep	Forwards the message without changing Option 82.
	Replace	Forwards the message after replacing the original Option 82 with the Option 82 padded according to the configured padding format, padding content, and code type.
No Option 82	N/A	Forwards the message after adding Option 82 padded according to the configured padding format, padding content, and code type.

DHCP relay agent tasks at a glance

To configure a DHCP relay agent, perform the following tasks:

1. Enabling DHCP
2. Enabling the DHCP relay agent on an interface
3. Specifying DHCP servers
4. (Optional.) Specifying a DHCP relay address pool for DHCP clients
5. (Optional.) Configuring the DHCP relay agent security features

6. (Optional.) Configuring the DHCP relay agent to release an IP address
7. (Optional.) Configuring DHCP relay agent support for Option 82
8. (Optional.) Setting the DSCP value for DHCP packets sent by the DHCP relay agent
9. (Optional.) Specifying the DHCP relay agent address for the giaddr field
10. (Optional.) Specifying the source IP address for relayed DHCP requests
11. (Optional.) Configuring forwarding DHCP replies based on Option 82

Enabling DHCP

Restrictions and guidelines

You must enable DHCP to make other DHCP relay agent settings take effect.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enable DHCP.
   ```
   dhcp enable
   ```
   By default, DHCP is disabled.

Enabling the DHCP relay agent on an interface

About this task

With the DHCP relay agent enabled, an interface forwards incoming DHCP requests to a DHCP server.

An IP address pool that contains the IP address of the DHCP relay interface must be configured on the DHCP server. Otherwise, the DHCP clients connected to the relay agent cannot obtain correct IP addresses.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enter interface view.
   ```
   interface interface-type interface-number
   ```
3. Enable the DHCP relay agent.
   ```
   dhcp select relay
   ```
   By default, when DHCP is enabled, an interface operates in the DHCP server mode.

Specifying DHCP servers

Specifying DHCP servers on a relay agent

About this task

To improve availability, you can specify several DHCP servers on the DHCP relay agent. When the interface receives request messages from clients, the relay agent forwards them to all DHCP servers.

Restrictions and guidelines

The IP address of any specified DHCP server must not reside on the same subnet as the IP address of the relay interface. Otherwise, the clients might fail to obtain IP addresses.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify a DHCP server address on the relay agent.

   `dhcp relay server-address` *`ip-address`*

   By default, no DHCP server address is specified on the relay agent.

   To specify multiple DHCP server addresses, repeat this step. You can specify a maximum of eight DHCP servers.

Specifying DHCP servers in a DHCP relay address pool

About this task

DHCP address pools created on a DHCP relay agent are called DHCP relay address pools. You can create a relay address pool and specify DHCP servers in this address pool. This feature allows DHCP clients of the same type to obtain IP addresses and other configuration parameters from the DHCP servers specified in the matching DHCP relay address pool.

It applies to scenarios where the DHCP relay agent connects to clients of the same access type but classified into different types by their locations. In this case, the relay interface typically has no IP address configured. You can use the **`gateway-list`** command to specify gateway addresses for clients matching the same DHCP relay address pool and bind the gateway addresses to the device's MAC address. Example network is the IPoE network.

Upon receiving a DHCP DISCOVER or REQUEST from a client that matches a DHCP relay address pool, the relay agent processes the packet as follows:

- Fills the **giaddr** field of the packet with a specified gateway address.
- Forwards the packet to all DHCP servers in the matching DHCP relay address pool.

The DHCP servers select a DHCP address pool according to the gateway address.

Restrictions and guidelines

If PPPoE users are in the network, follow these restrictions and guidelines when you configure the DHCP relay address pool:

- Enable the DHCP relay agent to record DHCP relay entries by using the **`dhcp relay client-information record`** command. When a PPPoE user goes offline, the DHCP relay agent can find a matching relay entry and send a DHCP-RELEASE message to the DHCP server. This mechanism ensures the DHCP server is aware of the releasing of the IP address in a timely manner.
- The **`remote-server`** command also configures the device as a DHCP relay agent. You do not need to enable the DHCP relay agent by using the **`dhcp select relay`** command.

Procedure

1. Enter system view.

   `system-view`

2. Create a DHCP relay address pool and enter its view.

   `dhcp server ip-pool` *`pool-name`*

3. Specify gateways in the DHCP relay address pool.

   `gateway-list` *`ip-address`*`&<1-64>` [**`export-route`**]

   By default, no gateway address is specified.

4. Specify DHCP servers in the DHCP relay address pool.

   `remote-server` *`ip-address`*`&<1-8>`

   By default, no DHCP server is specified in the DHCP relay address pool.

   You can specify a maximum of eight DHCP servers in one DHCP relay address pool for high availability.

Specifying a DHCP relay address pool for DHCP clients

About this task

After you configure multiple DHCP relay address pools on a DHCP relay agent, you can specify these pools on an interface. To match DHCP clients based on options, you can define option settings when you specify the relay address pools.

Procedure

1. Enter system view.

   `system-view`

2. Create a DHCP relay address pool and enter its view.

   `dhcp server ip-pool` *`pool-name`*

   By default, no DHCP relay address pools exist.

3. Specify DHCP servers in the DHCP relay address pool.

   `remote-server` *`ip-address`*&<1-8>

   By default, no DHCP server is specified in the DHCP relay address pool.

4. Specify gateway addresses for the clients matching the DHCP relay address pool.

   `gateway-list` *`ip-address`*&<1-64>

   By default, no gateway address is specified.

Configuring the DHCP relay agent security features

Enabling the DHCP relay agent to record relay entries

About this task

Perform this task to enable the DHCP relay agent to automatically record clients' IP-to-MAC bindings (relay entries) after they obtain IP addresses through DHCP.

Some security features use the relay entries to check incoming packets and block packets that do not match any entry. In this way, illegal hosts are not able to access external networks through the relay agent. Examples of the security features are ARP address check and authorized ARP.

Rustications and guidelines

The DHCP relay agent does not record IP-to-MAC bindings for DHCP clients running on synchronous/asynchronous serial interfaces.

Procedure

1. Enter system view.

   `system-view`

2. Enable the relay agent to record relay entries.

   `dhcp relay client-information record`

   By default, the relay agent does not record relay entries.

Enabling periodic refresh of dynamic relay entries

About this task

A DHCP client unicasts a DHCP-RELEASE message to the DHCP server to release its IP address. The DHCP relay agent conveys the message to the DHCP server and does not remove the IP-to-MAC entry of the client.

With this feature, the DHCP relay agent uses the IP address of a relay entry to periodically send a DHCP-REQUEST message to the DHCP server.

The relay agent maintains the relay entries depending on what it receives from the DHCP server:

- If the server returns a DHCP-ACK message or does not return any message within an interval, the DHCP relay agent removes the relay entry. In addition, upon receiving the DHCP-ACK message, the relay agent sends a DHCP-RELEASE message to release the IP address.
- If the server returns a DHCP-NAK message, the relay agent keeps the relay entry.

Procedure

1. Enter system view.

   `system-view`

2. Enable periodic refresh of dynamic relay entries.

   `dhcp relay client-information refresh enable`

   By default, periodic refresh of dynamic relay entries is enabled.

3. (Optional.) Set the refresh interval.

   `dhcp relay client-information refresh` `[` **`auto`** `|` **`interval`** *`interval`* `]`

   By default, the refresh interval is **`auto`**, which is calculated based on the number of total relay entries.

Enabling DHCP starvation attack protection

About this task

A DHCP starvation attack occurs when an attacker constantly sends forged DHCP requests using different MAC addresses in the **chaddr** field to a DHCP server. This exhausts the IP address resources of the DHCP server so legitimate DHCP clients cannot obtain IP addresses. The DHCP server might also fail to work because of exhaustion of system resources. The following methods are available to relieve or prevent such attacks.

- To relieve a DHCP starvation attack that uses DHCP packets encapsulated with different source MAC addresses, you can use one of the following methods:
  - Limit the number of ARP entries that a Layer 3 interface can learn.
  - Set the MAC learning limit for a Layer 2 port, and disable unknown frame forwarding when the MAC learning limit is reached.
- To prevent a DHCP starvation attack that uses DHCP requests encapsulated with the same source MAC address, you can enable MAC address check on the DHCP relay agent. The DHCP relay agent compares the **chaddr** field of a received DHCP request with the source MAC

address in the frame header. If they are the same, the DHCP relay agent forwards the request to the DHCP server. If not, the relay agent discards the request.

Enable MAC address check only on the DHCP relay agent directly connected to the DHCP clients. A DHCP relay agent changes the source MAC address of DHCP packets before sending them.

A MAC address check entry has an aging time. When the aging time expires, both of the following occur:

- The entry ages out.
- The DHCP relay agent rechecks the validity of DHCP requests sent from the MAC address in the entry.

Procedure

1. Enter system view.

   system-view

2. Set the aging time for MAC address check entries.

   dhcp relay check mac-address aging-time *time*

   The default aging time is 30 seconds.

   This command takes effect only after you execute the **dhcp relay check mac-address** command.

3. Enter the interface view.

   interface *interface-type interface-number*

4. Enable MAC address check.

   dhcp relay check mac-address

   By default, MAC address check is disabled.

Enabling DHCP server proxy on the DHCP relay agent

About this task

The DHCP server proxy feature isolates DHCP servers from DHCP clients and protects DHCP servers against attacks.

Upon receiving a response from the server, the DHCP server proxy modifies the server's IP address as the relay interface's IP address before sending out the response. The DHCP client takes the DHCP relay agent as the DHCP server.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable DHCP relay agent and DHCP server proxy on the interface.

   dhcp select relay proxy

   By default, the interface operates in DHCP server mode after DHCP is enabled.

Enabling client offline detection on the DHCP relay agent

About this task

The client offline detection on the DHCP relay agent detects the user online status based on the ARP entry aging. When an ARP entry ages out, the DHCP client offline detection feature deletes the relay entry for the IP address and sends a RELEASE message to the DHCP server.

Restrictions and guidelines

The feature does not function if an ARP entry is manually deleted.

Procedure

1. Enter system view.

   `system-view`

2. Enable the relay agent to record relay entries.

   `dhcp relay client-information record`

   By default, the relay agent does not record relay entries.

   Without relay entries, client offline detection cannot function correctly.

3. Enter interface view.

   `interface` *`interface-type interface-number`*

4. Enable the DHCP relay agent.

   `dhcp select relay`

   By default, when DHCP is enabled, an interface operates in the DHCP server mode.

5. Enable client offline detection.

   `dhcp client-detect`

   By default, client offline detection is disabled on the DHCP relay agent.

Configuring the DHCP relay agent to release an IP address

About this task

Configure the relay agent to release the IP address for a relay entry. The relay agent sends a DHCP-RELEASE message to the server and meanwhile deletes the relay entry. Upon receiving the DHCP-RELEASE message, the DHCP server releases the IP address.

This command can release only the IP addresses in the recorded relay entries.

Procedure

1. Enter system view.

   `system-view`

2. Configure the DHCP relay agent to release an IP address.

   `dhcp relay release ip` *`ip-address`* `[` **`vpn-instance`** *`vpn-instance-name`* `]`

Configuring DHCP relay agent support for Option 82

To support Option 82, you must perform related configuration on both the DHCP server and relay agent. For DHCP server Option 82 configuration, see "Enabling handling of Option 82."

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the relay agent to handle Option 82.

   dhcp relay information enable

   By default, handling of Option 82 is disabled.

4. (Optional.) Configure the strategy for handling DHCP requests that contain Option 82.

   dhcp relay information strategy { **drop** | **keep** | **replace** }

   By default, the handling strategy is **replace**.

   If the handling strategy is **replace**, configure a padding mode and a padding format for Option 82. If the handling strategy is **keep** or **drop**, you do not need to configure a padding mode or padding format for Option 82.

5. (Optional.) Configure the padding mode and padding format for the Circuit ID sub-option.

   dhcp relay information circuit-id { **bas** [**sub-interface-vlan**] | **string** *circuit-id* | { **normal** | **verbose** [**node-identifier** { **mac** | **sysname** | **user-defined** *node-identifier* }] [**interface**] [**sub-interface-vlan**] } [**format** { **ascii** | **hex** }] }

   By default, the padding mode for Circuit ID sub-option is **normal**, and the padding format is **hex**.

   The device name (**sysname**) must not include spaces if it is configured as the padding content for sub-option 1. Otherwise, the DHCP relay agent will fail to add or replace Option 82.

6. (Optional.) Configure the padding mode and padding format for the Remote ID sub-option.

   dhcp relay information remote-id {**normal** [**format** { **ascii** | **hex** }] | **string** *remote-id* | **sysname** }

   By default, the padding mode for the Remote ID sub-option is **normal**, and the padding format is **hex**.

Setting the DSCP value for DHCP packets sent by the DHCP relay agent

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.

   system-view

2. Set the DSCP value for DHCP packets sent by the DHCP relay agent.

   dhcp dscp *dscp-value*

   By default, the DSCP value in DHCP packets sent by the DHCP relay agent is 56.

Specifying the DHCP relay agent address for the **giaddr** field

Manually specifying the DHCP relay agent address for the **giaddr** field

About this task

This task allows you to specify the IP addresses to be encapsulated to the **giaddr** field of the DHCP requests. If you do not specify any DHCP relay agent address, the primary IP address of the DHCP relay interface is encapsulated to the **giaddr** field of DHCP requests.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the DHCP relay agent address to be encapsulated in relayed DHCP requests.

   dhcp relay gateway *ip-address*

   By default, the primary IP address of the DHCP relay interface is encapsulated in the relayed DHCP requests.

Configuring smart relay to specify the DHCP relay agent address for the **giaddr** field

About this task

By default, the relay agent only encapsulates the primary IP address to the **giaddr** field of all requests before relaying them to the DHCP server. The DHCP server then selects an IP address on the same subnet as the address in the **giaddr** filed. If no assignable addresses on the subnet are available, the DHCP server does not assign any IP address. The DHCP smart relay feature is introduced to allow the DHCP relay agent to encapsulate secondary IP addresses when the DHCP server does not send back a DHCP-OFFER message.

The relay agent initially encapsulates its primary IP address to the **giaddr** field before forwarding a request to the DHCP server. If no DHCP-OFFER is received, the relay agent allows the client to send a maximum of two requests to the DHCP server by using the primary IP address. If no DHCP-OFFER is returned after two retries, the relay agent switches to a secondary IP address. If the DHCP server still does not respond, the next secondary IP address is used. After the secondary IP addresses are all tried and the DHCP server does not respond, the relay agent repeats the process by starting from the primary IP address.

Procedure

1. Enter system view.

   system-view

2. Enable the DHCP smart relay feature.

```
dhcp smart-relay enable
```

By default, the DHCP smart relay feature is disabled.

Specifying the source IP address for relayed DHCP requests

About this task

This task is required if multiple relay interfaces share the same IP address or if a relay interface does not have routes to DHCP servers. You can specify an IP address or the IP address of another interface, typically the loopback interface, on the DHCP relay agent as the source IP address for DHCP requests. The relay interface inserts the source IP address in the source IP address field as well as the **giaddr** field in DHCP requests.

If multiple relay interfaces share the same IP address, you must also configure the relay interface to support Option 82. Upon receiving a DHCP request, the relay interface inserts the subnet information in sub-option 5 in Option 82. The DHCP server assigns an IP address according to sub-option 5. The DHCP relay agent looks up the output interface in the MAC address table to forward the DHCP reply.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify the source IP address for DHCP requests.

   ```
   dhcp relay source-address { ip-address | gateway | relay-interface }
   ```

   The relay agent chooses the default source IP address for relayed requests depending on whether its server-side interface and the DHCP server belong to the same VPN instance:

   - If they belong to the same VPN instance, the relay agent uses the IP address of the output interface for relayed requests as the default source IP address.
   - If they belong to different VPN instances, the relay agent uses the lowest IP address that is in the same VPN instance as the DHCP server as the default source address.

Configuring forwarding DHCP replies based on Option 82

About this task

Configure this feature if the DHCP relay agent is required to forward DHCP replies to DHCP clients based on Option 82.

For example, an IPRAN network has a primary gateway and a secondary gateway. An L3VE interface is configured as the relay interface on each gateway. Multiple L2VE subinterfaces are configured to receive packets. One L2VE subinterface corresponds to one PW. Only the primary gateway receives DHCP requests, but both the primary and secondary gateways might receive DHCP replies. The primary gateway can forward DHCP replies based on locally recorded user information, but the secondary gateway cannot. The secondary gateway can only forward DHCP replies to all PWs.

To enable the secondary gateway to forward a DHCP reply only to the intended PW, perform the following tasks:

- Configure the **dhcp relay information enable** and **dhcp relay information circuit-id** (with **sub-interface-vlan** specified) commands on the primary gateway. Then, when the primary gateway receives a DHCP request, it adds Option 82 to the reply and records the VLAN ID of the L2VE subinterface.
- Configure the **dhcp relay information enable**, **dhcp relay information circuit-id** (with **sub-interface-vlan** specified), and **dhcp relay forward reply by-option82** commands on the secondary gateway. Then, when the secondary gateway receives a DHCP reply, it resolves Option 82, records the VLAN ID of the L2VE subinterface, and forwards the reply to the PW.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the relay agent to handle Option 82.

   dhcp relay information enable

   By default, handling of Option 82 is disabled.

4. Configure the padding mode and padding format for the Circuit ID sub-option.

   dhcp relay information circuit-id { **bas** [**sub-interface-vlan**] | **string** *circuit-id* | { **normal** | **verbose** [**node-identifier** { **mac** | **sysname** | **user-defined** *node-identifier* }] [**interface**] } [**sub-interface-vlan**] [**format** { **ascii** | **hex** }] }

   By default, the padding mode for Circuit ID sub-option is **normal**, and the padding format is **hex**.

   The device name (**sysname**) must not include spaces if it is configured as the padding content for sub-option 1. Otherwise, the DHCP relay agent will fail to add or replace Option 82.

5. Configure the DHCP relay agent to forward DHCP replies based on Option 82.

   dhcp relay forward reply by-option82

   By default, the DHCP relay agent does not forward DHCP replies based on Option 82.

Display and maintenance commands for DHCP relay agent

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display MAC address check entries on the DHCP relay agent.	**display dhcp relay check mac-address**
Display relay entries on the DHCP relay agent.	**display dhcp relay client-information** [**interface** *interface-type interface-number* | **ip** *ip-address* [**vpn-instance** *vpn-instance-name*]]
Display Option 82 configuration information on the DHCP relay agent.	**display dhcp relay information** [**interface** *interface-type interface-number*]
Display information about DHCP servers	**display dhcp relay server-address**

Task	Command
on an interface.	[**interface** *interface-type* *interface-number*]
Display packet statistics on the DHCP relay agent.	**display dhcp relay statistics** [**interface** *interface-type interface-number*]
Clear relay entries on the DHCP relay agent.	**reset dhcp relay client-information** [**interface** *interface-type* *interface-number* | **ip** *ip-address* [**vpn-instance** *vpn-instance-name*]]
Clear packet statistics on the DHCP relay agent.	**reset dhcp relay statistics** [**interface** *interface-type interface-number*]

Troubleshooting DHCP relay agent configuration

Failure of DHCP clients to obtain configuration parameters through the DHCP relay agent

Symptom

DHCP clients cannot obtain configuration parameters through the DHCP relay agent.

Solution

Some problems might occur with the DHCP relay agent or server configuration.

To locate the problem, enable debugging and execute the **display** command on the DHCP relay agent to view the debugging information and interface state information.

Check that:

- DHCP is enabled on the DHCP server and relay agent.
- The DHCP server has an address pool on the same subnet as the DHCP clients.
- The DHCP server and DHCP relay agent can reach each other.
- The DHCP server address specified on the DHCP relay interface connected to the DHCP clients is correct.

Configuring the DHCP client

About DHCP client

With DHCP client enabled, an interface uses DHCP to obtain configuration parameters from the DHCP server, for example, an IP address.

Restrictions and guidelines: DHCP client configuration

The DHCP client configuration is supported only on Layer 3 Ethernet interfaces (or subinterfaces), VLAN interfaces, and Layer 3 aggregate interfaces.

DHCP client tasks at a glance

To configure a DHCP client, perform the following tasks:

1. Enabling the DHCP client on an interface
2. Configuring a DHCP client ID for an interface

   Perform this task if the DHCP client uses the client ID to obtain IP addresses.
3. (Optional.) Enabling duplicated address detection
4. (Optional.) Setting the DSCP value for DHCP packets sent by the DHCP client

Enabling the DHCP client on an interface

Restrictions and guidelines

- If the number of IP address request failures reaches the system-defined amount, the DHCP client-enabled interface uses a default IP address.
- An interface can be configured to acquire an IP address in multiple ways. The new configuration overwrites the old.
- Secondary IP addresses cannot be configured on an interface that is enabled with the DHCP client.
- If the interface obtains an IP address on the same segment as another interface on the device, the interface does not use the assigned address. Instead, it requests a new IP address from the DHCP server.

Procedure

1. Enter system view.

   `system-view`
2. Enter interface view.

   `interface` *`interface-type interface-number`*
3. Configure an interface to use DHCP for IP address acquisition.

   `ip address dhcp-alloc`

   By default, an interface does not use DHCP for IP address acquisition.

Configuring a DHCP client ID for an interface

About this task

A DHCP client ID is added to the DHCP option 61 to uniquely identify a DHCP client. A DHCP server can assign IP addresses to clients based on their DHCP client IDs.

DHCP client ID includes an ID type and a type value. Each ID type has a fixed type value. You can specify a DHCP client ID by using one of the following methods:

- Use an ASCII string as the client ID. If an ASCII string is used, the type value is 00.
- Use a hexadecimal number as the client ID. If a hexadecimal number is used, the type value is the first two characters in the number.
- Use the MAC address of an interface to generate a client ID. If this method is used, the type value is 01.

The type value of a DHCP client ID can be displayed by the **`display dhcp server ip-in-use`** or **`display dhcp client`** command.

Restrictions and guidelines

Make sure the ID for each DHCP client is unique.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure a DHCP client ID for the interface.

   `dhcp client identifier` { **`ascii`** *`ascii-string`* | **`hex`** *`hex-string`* | **`mac`** *`interface-type interface-number`* }

   By default, an interface generates the DHCP client ID based on its MAC address. If the interface has no MAC address, it uses the MAC address of the first Ethernet interface to generate its client ID.

Enabling duplicated address detection

About this task

DHCP client detects IP address conflict through ARP packets. An attacker can act as the IP address owner to send an ARP reply. The spoofing attack makes the client unable to use the IP address assigned by the server. As a best practice, disable duplicate address detection when ARP attacks exist on the network.

Procedure

1. Enter system view.

   `system-view`

2. Enable duplicate address detection.

   `dhcp client dad enable`

   By default, the duplicate address detection feature is enabled on an interface.

Setting the DSCP value for DHCP packets sent by the DHCP client

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.

   `system-view`

2. Set the DSCP value for DHCP packets sent by the DHCP client.

   `dhcp client dscp` *`dscp-value`*

   By default, the DSCP value in DHCP packets sent by the DHCP client is 56.

Display and maintenance commands for DHCP client

Execute **`display`** command in any view.

Task	Command
Display DHCP client information.	**`display dhcp client`** [**`verbose`**] [**`interface`** *`interface-type interface-number`*]

Configuring the BOOTP client

About BOOTP client

BOOTP client application

An interface that acts as a BOOTP client can use BOOTP to obtain information (such as IP address) from the BOOTP server.

To use BOOTP, an administrator must configure a BOOTP parameter file for each BOOTP client on the BOOTP server. The parameter file contains information such as MAC address and IP address of a BOOTP client. When a BOOTP client sends a request to the BOOTP server, the BOOTP server searches for the BOOTP parameter file and returns the corresponding configuration information.

BOOTP is usually used in relatively stable environments. In network environments that change frequently, DHCP is more suitable.

Because a DHCP server can interact with a BOOTP client, you can use the DHCP server to assign an IP address to the BOOTP client. You do not need to configure a BOOTP server. The DHCP server will assign an IP address to the BOOTP client based on the IP address allocation sequence.

Obtaining an IP address dynamically

A BOOTP client dynamically obtains an IP address from a BOOTP server as follows:

1. The BOOTP client broadcasts a BOOTP request, which contains its own MAC address.
2. Upon receiving the request, the BOOTP server searches the configuration file for the IP address and other information according to the BOOTP client's MAC address.
3. The BOOTP server returns a BOOTP response to the BOOTP client.
4. The BOOTP client obtains the IP address from the received response.

A DHCP server can take the place of the BOOTP server in the following dynamic IP address acquisition.

Protocols and standards

- RFC 951, *Bootstrap Protocol (BOOTP)*
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
- RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*

Configuring an interface to use BOOTP for IP address acquisition

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   BOOTP client configuration applies only to Layer 3 Ethernet interfaces (including subinterfaces), Layer 3 aggregate interfaces, and VLAN interfaces.

3. Configure an interface to use BOOTP for IP address acquisition.

   ip address bootp-alloc

   By default, an interface does not use BOOTP for IP address acquisition.

Display and maintenance commands for BOOTP client

Execute **display** command in any view.

Task	Command
Display BOOTP client information.	**display bootp client** [**interface** *interface-type interface-number*]

Contents

DHCPv6 overview

DHCPv6 provides a framework to assign IPv6 prefixes, IPv6 addresses, and other configuration parameters to hosts.

DHCPv6 address/prefix assignment

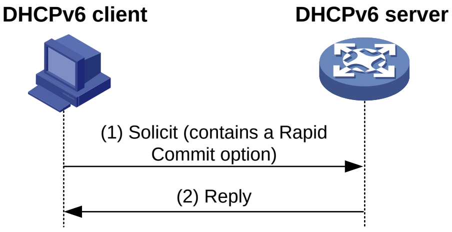
An address/prefix assignment process involves two or four messages.

Rapid assignment involving two messages

As shown in Figure 1, rapid assignment operates in the following steps:

1. The DHCPv6 client sends to the DHCPv6 server a Solicit message that contains a Rapid Commit option to prefer rapid assignment.
2. If the DHCPv6 server supports rapid assignment, it responds with a Reply message containing the assigned IPv6 address/prefix and other configuration parameters. If the DHCPv6 server does not support rapid assignment, Assignment involving four messages is performed.

Figure 1 Rapid assignment involving two messages

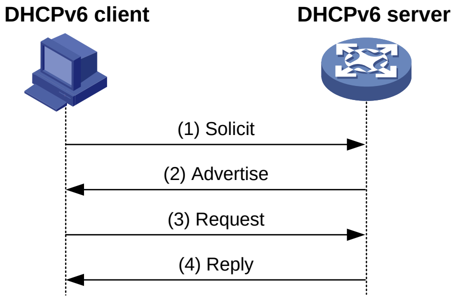


Assignment involving four messages

As shown in Figure 2, four-message assignment operates using the following steps:

1. The DHCPv6 client sends a Solicit message to request an IPv6 address/prefix and other configuration parameters.
2. The DHCPv6 server responds with an Advertise message that contains the assignable address/prefix and other configuration parameters if either of the following conditions exists:
    - The Solicit message does not contain a Rapid Commit option.
    - The DHCPv6 server does not support rapid assignment even though the Solicit message contains a Rapid Commit option.
3. The DHCPv6 client might receive multiple Advertise messages offered by different DHCPv6 servers. It selects an offer according to the receiving sequence and server priority, and sends a Request message to the selected server for confirmation.
4. The DHCPv6 server sends a Reply message to the client, confirming that the address/prefix and other configuration parameters are assigned to the client.

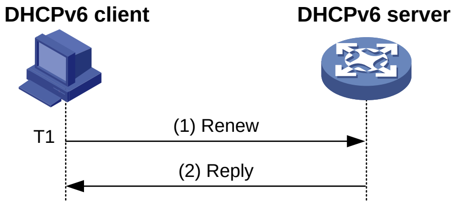
Figure 2 Assignment involving four messages



Address/prefix lease renewal

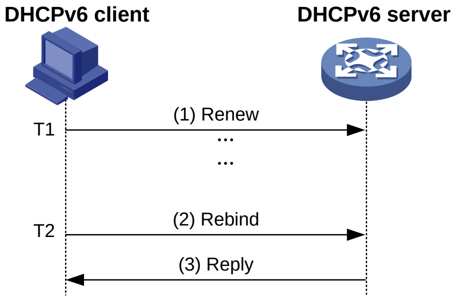
An IPv6 address/prefix assigned by a DHCPv6 server has a valid lifetime. After the valid lifetime expires, the DHCPv6 client cannot use the IPv6 address/prefix. To use the IPv6 address/prefix, the DHCPv6 client must renew the lease time.

Figure 3 Using the Renew message for address/prefix lease renewal



As shown in Figure 3, at T1, the DHCPv6 client sends a Renew message to the DHCPv6 server. The recommended value of T1 is half the preferred lifetime. The DHCPv6 server responds with a Reply message, informing the client whether the lease is renewed.

Figure 4 Using the Rebind message for address/prefix lease renewal



As shown in Figure 4:

- If the DHCPv6 client does not receive a response from the DHCPv6 server after sending a Renew message at T1, it multicasts a Rebind message to all DHCPv6 servers at T2. Typically, the value of T2 is 0.8 times the preferred lifetime.
- The DHCPv6 server responds with a Reply message, informing the client whether the lease is renewed.
- If the DHCPv6 client does not receive a response from any DHCPv6 server before the valid lifetime expires, the client stops using the address/prefix.

For more information about the valid lifetime and the preferred lifetime, see "Configuring basic IPv6 settings."

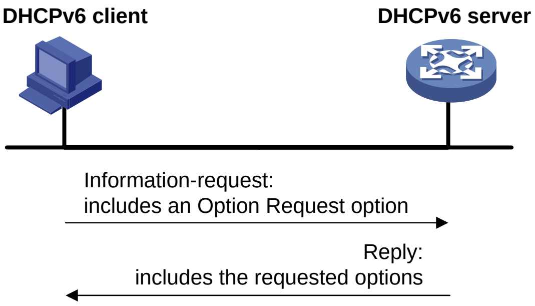
Stateless DHCPv6

Stateless DHCPv6 enables a device that has obtained an IPv6 address/prefix to get other configuration parameters from a DHCPv6 server.

The device performs stateless DHCPv6 if an RA message with the following flags is received from the router during stateless address autoconfiguration:

- The managed address configuration flag (M flag) is set to 0.
- The other stateful configuration flag (O flag) is set to 1.

Figure 5 Stateless DHCPv6 operation


DHCPv6 client
DHCPv6 server
Information-request:
includes an Option Request option
Reply:
includes the requested options

As shown in Figure 5, stateless DHCPv6 operates in the following steps:

1. The DHCPv6 client sends an Information-request message to the multicast address of all DHCPv6 servers and DHCPv6 relay agents. The Information-request message contains an Option Request option that specifies the requested configuration parameters.
2. The DHCPv6 server returns to the client a Reply message containing the requested configuration parameters.
3. The client checks the Reply message. If the obtained configuration parameters match those requested in the Information-request message, the client uses these parameters to complete configuration. If not, the client ignores the configuration parameters. If the client receives multiple replies with configuration parameters matching those requested in the Information-request message, it uses the first received reply.

DHCPv6 options

Option 18

Option 18, also called the interface-ID option, is used by the DHCPv6 relay agent to determine the interface to use to forward RELAY-REPLY message.

The DHCPv6 snooping device adds Option 18 to the received DHCPv6 request message before forwarding it to the DHCPv6 server. The server then assigns IP address to the client based on the client information in Option 18.

Figure 6 Option 18 format

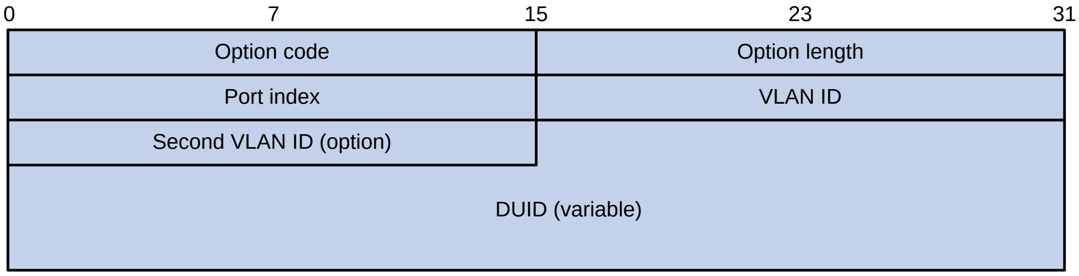


Figure 6 shows the Option 18 format, which includes the following fields:

- **Option code**—Option code. The value is 18.
- **Option length**—Size of the option data.
- **Port index**—Port that receives the DHCPv6 request from the client.
- **VLAN ID**—ID of the outer VLAN.
- **Second VLAN ID**—ID of the inner VLAN. This field is optional. If the received DHCPv6 request does not contain a second VLAN, Option 18 also does not contain it.
- **DUID**—DUID of the DHCPv6 client.

Option 37

Option 37, also called the remote-ID option, is used to identify the client.

The DHCPv6 snooping device adds Option 37 to the received DHCPv6 request message before forwarding it to the DHCPv6 server. This option provides client information about address allocation.

Figure 7 Option 37 format

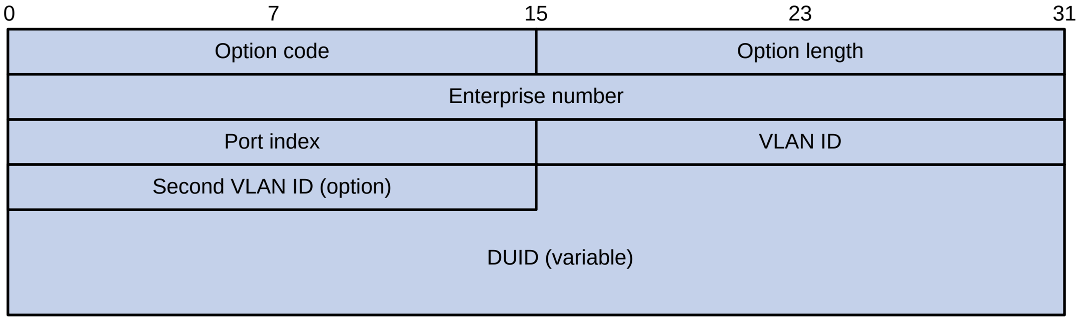


Figure 7 shows the Option 37 format, which includes the following fields:

- **Option code**—Option code. The value is 37.
- **Option length**—Size of the option data.
- **Enterprise number**—Enterprise number.
- **Port index**—Port that receives the DHCPv6 request from the client.
- **VLAN ID**—ID of the outer VLAN.
- **Second VLAN ID**—ID of the inner VLAN. This field is optional. If the received DHCPv6 request does not contain a second VLAN, Option 37 also does not contain it.
- **DUID**—DUID of the DHCPv6 client.

Protocols and standards

- RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*
- RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*
- RFC 2462, *IPv6 Stateless Address Autoconfiguration*
- RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*

Configuring the DHCPv6 server

About DHCPv6 server

A DHCPv6 server can assign IPv6 addresses, IPv6 prefixes, and other configuration parameters to DHCPv6 clients.

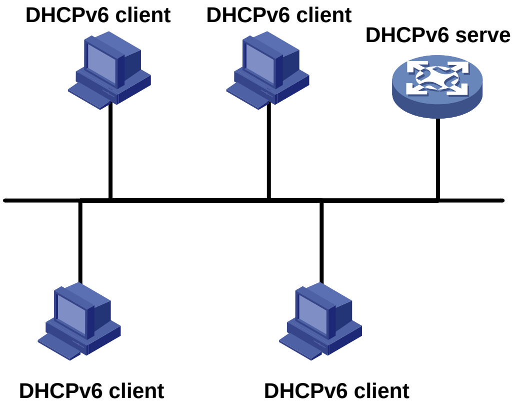
IPv6 address assignment

As shown in Figure 8, the DHCPv6 server assigns IPv6 addresses, domain name suffixes, DNS server addresses, and other configuration parameters to DHCPv6 clients.

The IPv6 addresses assigned to the clients include the following types:

- **Temporary IPv6 addresses**—Frequently changed without lease renewal.
- **Non-temporary IPv6 addresses**—Correctly used by DHCPv6 clients, with lease renewal.

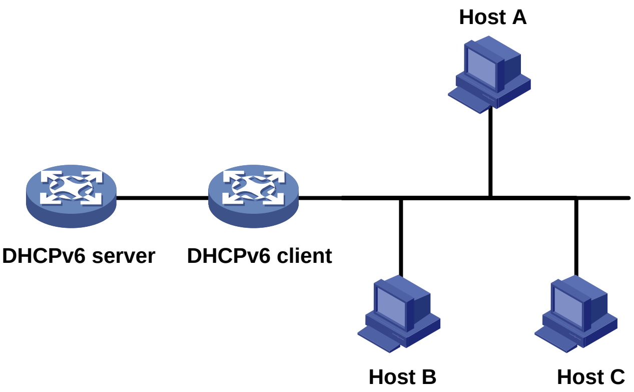
Figure 8 IPv6 address assignment



IPv6 prefix assignment

As shown in Figure 9, the DHCPv6 server assigns an IPv6 prefix to the DHCPv6 client. The client advertises the prefix information in a multicast RA message so that hosts on the subnet can automatically configure their IPv6 addresses by using the prefix.

Figure 9 IPv6 prefix assignment



Concepts

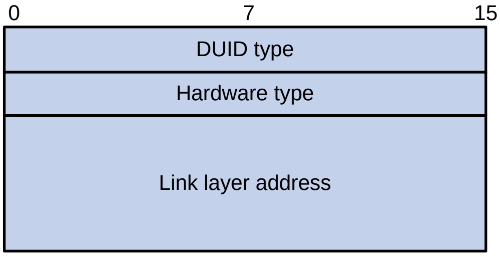
Multicast addresses used by DHCPv6

DHCPv6 uses the multicast address FF05::1:3 to identify all site-local DHCPv6 servers. It uses the multicast address FF02::1:2 to identify all link-local DHCPv6 servers and relay agents.

DUID

A DHCP unique identifier (DUID) uniquely identifies a DHCPv6 device (DHCPv6 client, server, or relay agent). A DHCPv6 device adds its DUID in a sent packet.

Figure 10 DUID-LL format



The device supports the DUID format based on link-layer address (DUID-LL) defined in RFC 3315. Figure 10 shows the DUID-LL format, which includes the following fields:

- **DUID type**—The device supports the DUID type of DUID-LL with the value of 0x0003.
- **Hardware type**—The device supports the hardware type of Ethernet with the value of 0x0001.
- **Link layer address**—Takes the value of the bridge MAC address of the device.

IA

Identified by an IAID, an identity association (IA) provides a construct through which a client manages the obtained addresses, prefixes, and other configuration parameters. A client can have multiple IAs, for example, one for each of its interfaces.

IAID

An IAID uniquely identifies an IA. It is chosen by the client and must be unique on the client.

PD

The DHCPv6 server creates a prefix delegation (PD) for each assigned prefix to record the following details:

- IPv6 prefix.
- Client DUID.
- IAID.
- Valid lifetime.
- Preferred lifetime.
- Lease expiration time.
- IPv6 address of the requesting client.

DHCPv6 address pool

The DHCP server selects IPv6 addresses, IPv6 prefixes, and other parameters from an address pool, and assigns them to the DHCP clients.

Address allocation mechanisms

DHCPv6 supports the following address allocation mechanisms:

- **Static address allocation**—To implement static address allocation for a client, create a DHCPv6 address pool, and manually bind the DUID and IAID of the client to an IPv6 address in the DHCPv6 address pool. When the client requests an IPv6 address, the DHCPv6 server assigns the IPv6 address in the static binding to the client.
- **Dynamic address allocation**—To implement dynamic address allocation for clients, create a DHCPv6 address pool, specify a subnet for the pool, and divide the subnet into temporary and non-temporary IPv6 address ranges. Upon receiving a DHCP request, the DHCPv6 server selects an IPv6 address from the temporary or non-temporary IPv6 address range based on the address type in the client request.

Prefix allocation mechanisms

DHCPv6 supports the following prefix allocation mechanisms:

- **Static prefix allocation**—To implement static prefix allocation for a client, create a DHCPv6 address pool, and manually bind the DUID and IAID of the client to an IPv6 prefix in the DHCPv6 address pool. When the client requests an IPv6 prefix, the DHCPv6 server assigns the IPv6 prefix in the static binding to the client.
- **Dynamic prefix allocation**—To implement dynamic prefix allocation for clients, create a DHCPv6 address pool and a prefix pool, specify a subnet for the address pool, and apply the prefix pool to the address pool. Upon receiving a DHCP request, the DHCPv6 server dynamically selects an IPv6 prefix from the prefix pool in the address pool.

Address pool selection

The DHCPv6 server observes the following principles when selecting an IPv6 address or prefix for a client:

1. If there is an address pool where an IPv6 address is statically bound to the DUID or IAID of the client, the DHCPv6 server selects this address pool. It assigns the statically bound IPv6 address or prefix and other configuration parameters to the client.
2. If the receiving interface has a DHCP policy and the DHCP client matches a user class, the DHCP server selects the address pool that is bound to the matching user class. If no matching user class is found, the server assigns an IP address and other parameters from the default DHCP address pool. If no default address pool is specified or the default address pool does not have assignable IP addresses, the address assignment fails.
3. If the receiving interface has an address pool applied, the DHCP server selects an IPv6 address or prefix and other configuration parameters from this address pool.
4. If the above conditions are not met, the DHCPv6 server selects an address pool depending on the client location.
   - **Client on the same subnet as the server**—The DHCPv6 server compares the IPv6 address of the receiving interface with the subnets of all address pools. It selects the address pool with the longest-matching subnet.
   - **Client on a different subnet than the server**—The DHCPv6 server compares the IPv6 address of the DHCPv6 relay agent interface closest to the client with the subnets of all address pools. It also selects the address pool with the longest-matching subnet.

   To make sure IPv6 address allocation functions correctly, keep the subnet used for dynamic assignment consistent with the subnet where the interface of the DHCPv6 server or DHCPv6 relay agent resides.

IPv6 address/prefix allocation sequence

The DHCPv6 server selects an IPv6 address/prefix for a client in the following sequence:

1. IPv6 address/prefix statically bound to the client's DUID and IAID and expected by the client.

2. IPv6 address/prefix statically bound to the client's DUID and IAID.
3. IPv6 address/prefix statically bound to the client's DUID and expected by the client.
4. IPv6 address/prefix statically bound to the client's DUID.
5. Assignable IPv6 address/prefix in the address pool/prefix pool expected by the client.
6. IPv6 address/prefix that was ever assigned to the client.
7. Assignable IPv6 address/prefix in the address pool/prefix pool.
8. IPv6 address/prefix that was a conflict or passed its lease duration. If no IPv6 address/prefix is assignable, the server does not respond.

If a client moves to another subnet, the DHCPv6 server selects an IPv6 address/prefix from the address pool that matches the new subnet.

Conflicted IPv6 addresses can be assigned to other DHCPv6 clients only after the addresses are in conflict for one hour.

DHCPv6 server tasks at a glance

To configure the DHCPv6 server, perform the following tasks:

1. Configuring the DHCPv6 server to assign IPv6 prefixes, IPv6 addresses, and other network parameters

   Choose the following tasks as needed:
   - Configuring IPv6 prefix assignment
   - Configuring IPv6 address assignment
   - Configuring network parameters assignment
2. Modifying the address pool selection method on the DHCPv6 server

   Choose the following tasks as needed:
   - Configuring the DHCPv6 server on an interface
   - Configuring a DHCPv6 policy for IPv6 address and prefix assignment
3. (Optional.) Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 server
4. (Optional.) Configuring DHCPv6 binding auto backup
5. (Optional.) Advertising subnets assigned to clients
6. (Optional.) Applying a DHCPv6 address pool to a VPN instance
7. (Optional.) Enabling DHCPv6 logging on the DHCPv6 server

Configuring IPv6 prefix assignment

About this task

Use the following methods to configure IPv6 prefix assignment:

- **Configure a static IPv6 prefix binding in an address pool**—If you bind a DUID and an IAID to an IPv6 prefix, the DUID and IAID in a request must match those in the binding before the DHCPv6 server can assign the IPv6 prefix to the DHCPv6 client. If you only bind a DUID to an IPv6 prefix, the DUID in the request must match the DUID in the binding before the DHCPv6 server can assign the IPv6 prefix to the DHCPv6 client.
- **Apply a prefix pool to an address pool**—The DHCPv6 server dynamically assigns an IPv6 prefix from the prefix pool in the address pool to a DHCPv6 client.

Restrictions and guidelines

When you configure IPv6 prefix assignment, follow these restrictions and guidelines:

- An IPv6 prefix can be bound to only one DHCPv6 client. You cannot modify bindings that have been created. To change the binding for a DHCPv6 client, you must delete the existing binding first.
- One address pool can have only one prefix pool applied. You cannot modify prefix pools that have been applied. To change the prefix pool for an address pool, you must remove the prefix pool application first.
- You can apply a prefix pool that has not been created to an address pool. The setting takes effect after the prefix pool is created.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Specify the IPv6 prefixes excluded from dynamic assignment.

   ipv6 dhcp server forbidden-prefix *start-prefix*/*prefix-len* [*end-prefix*/*prefix-len*] [**vpn-instance** *vpn-instance-name*]

   By default, no IPv6 prefixes in the prefix pool are excluded from dynamic assignment.

   If the excluded IPv6 prefix is in a static binding, the prefix still can be assigned to the client.

3. Create a prefix pool.

   ipv6 dhcp prefix-pool *prefix-pool-number* **prefix** { *prefix-number* | *prefix*/*prefix-len* } **assign-len** *assign-len* [**vpn-instance** *vpn-instance-name*]

   This step is required for dynamic prefix assignment.

   If you specify an IPv6 prefix by its ID, make sure the IPv6 prefix is in effect. Otherwise, the configuration does not take effect.

4. Enter DHCP address pool view.

   ipv6 dhcp pool *pool-name*

5. Specify an IPv6 subnet for dynamic assignment.

   network { *prefix/prefix-length* | **prefix** *prefix-number* [*sub-prefix/sub-prefix-length*] } [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*]

   By default, no IPv6 subnet is specified for dynamic assignment.

   The IPv6 subnets cannot be the same in different address pools.

   If you specify an IPv6 prefix by its ID, make sure the IPv6 prefix is in effect. Otherwise, the configuration does not take effect.

6. Configure the prefix assignment. Choose the options to configure as needed:
   - Configure a static prefix binding:

     static-bind prefix *prefix*/*prefix-len* **duid** *duid* [**iaid** *iaid*] [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*] [**description** *description-text*]

     By default, no static prefix binding is configured.

     To add multiple static IPv6 prefix bindings, repeat this step.
   - Apply the prefix pool to the address pool:

     prefix-pool *prefix-pool-number* [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*]

     By default, static or dynamic prefix assignment is not configured for an address pool.

Configuring IPv6 address assignment

About this task

Use one of the following methods to configure IPv6 address assignment:

- Configure a static IPv6 address binding in an address pool.

  If you bind a DUID and an IAID to an IPv6 address, the DUID and IAID in a request must match those in the binding before the DHCPv6 server can assign the IPv6 address to the requesting client. If you only bind a DUID to an IPv6 address, the DUID in a request must match the DUID in the binding before the DHCPv6 server can assign the IPv6 address to the requesting client.
- Specify a subnet and address ranges in an address pool.
  - **Non-temporary address assignment**—The server selects addresses from the non-temporary address range specified by the **`address range`** command. If no non-temporary address range is specified, the server selects addresses on the subnet specified by the **`network`** command.
  - **Temporary address assignment**—The server selects addresses from the temporary address range specified by the **`temporary address range`** command. If no temporary address range is specified in the address pool, the DHCPv6 server cannot assign temporary addresses to clients.

Restrictions and guidelines

- You can specify only one non-temporary address range and one temporary address range in an address pool.
- The address ranges specified by the **`address range`** and **`temporary address range`** commands must be on the subnet specified by the **`network`** command. Otherwise, the addresses are unassignable.
- An IPv6 address can be bound to only one DHCPv6 client. You cannot modify bindings that have been created. To change the binding for a DHCPv6 client, you must delete the existing binding first.
- Only one subnet can be specified in an address pool. If you use the **`network`** command multiple times in a DHCPv6 address pool, the most recent configuration takes effect. If you use this command to specify only new lifetimes, the settings do not affect existing leases. The IPv6 addresses assigned after the modification will use the new lifetimes.

Procedure

1. Enter system view.

   ```
   system-view
   ```
2. (Optional.) Specify the IPv6 addresses excluded from dynamic assignment.

   ```
   ipv6 dhcp server forbidden-address start-ipv6-address
   [ end-ipv6-address ] [ vpn-instance vpn-instance-name ]
   ```

   By default, all IPv6 addresses except for the DHCPv6 server's IP address in a DHCPv6 address pool are assignable.

   If the excluded IPv6 address is in a static binding, the address still can be assigned to the client.
3. Enter DHCPv6 address pool view.

   ```
   ipv6 dhcp pool pool-name
   ```
4. Specify an IPv6 subnet for dynamic assignment.

   ```
   network { prefix/prefix-length | prefix prefix-number
   [ sub-prefix/sub-prefix-length ] } [ preferred-lifetime
   preferred-lifetime valid-lifetime valid-lifetime ]
   ```

   By default, no IPv6 address subnet is specified.

The IPv6 subnets cannot be the same in different address pools.

If you specify an IPv6 prefix by its ID, make sure the IPv6 prefix is in effect. Otherwise, the configuration does not take effect.

5. (Optional.) Specify a non-temporary IPv6 address range.

   address range *start-ipv6-address end-ipv6-address* [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*]

   By default, no non-temporary IPv6 address range is specified, and all unicast addresses on the subnet are assignable.

6. (Optional.) Specify a temporary IPv6 address range.

   temporary address range *start-ipv6-address end-ipv6-address* [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*]

   By default, no temporary IPv6 address range is specified, and the DHCPv6 server cannot assign temporary IPv6 addresses.

7. (Optional.) Create a static binding.

   static-bind address *ipv6-address/addr-prefix-length* **duid** *duid* [**iaid** *iaid*] [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*] [**description** *description-text*]

   By default, no static binding is configured.

   To add more static bindings, repeat this step.

Configuring network parameters assignment

About network parameters assignment

In addition to IPv6 prefixes and IPv6 addresses, you can configure the following network parameters in an address pool:

- A maximum of eight DNS server addresses.
- One domain name.
- A maximum of eight SIP server addresses.
- A maximum of eight SIP server domain names.

You can configure network parameters on a DHCPv6 server by using one of the following methods:

- Configure network parameters in a DHCPv6 address pool.
- Configure network parameters in a DHCPv6 option group, and specify the option group for a DHCPv6 address pool.

Network parameters configured in a DHCPv6 address pool take precedence over those configured in a DHCPv6 option group.

Configuring network parameters in a DHCPv6 address pool

1. Enter system view.

   system-view

2. Enter DHCPv6 address pool view.

   ipv6 dhcp pool *pool-name*

3. Specify an IPv6 subnet for dynamic assignment.

network { *prefix/prefix-length* | **prefix** *prefix-number* [*sub-prefix/sub-prefix-length*] } [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*]

By default, no IPv6 subnet is specified.

The IPv6 subnets cannot be the same in different address pools.

If you specify an IPv6 prefix by its ID, make sure the IPv6 prefix is in effect. Otherwise, the configuration does not take effect.

4. Specify a DNS server address.

   dns-server *ipv6-address*

   By default, no DNS server address is specified.

5. Specify a domain name.

   domain-name *domain-name*

   By default, no domain name is specified.

6. Specify a SIP server address or domain name.

   sip-server { **address** *ipv6-address* | **domain-name** *domain-name* }

   By default, no SIP server address or domain name is specified.

7. Configure a self-defined DHCPv6 option.

   option *code* **hex** *hex-string*

   By default, no self-defined DHCPv6 option is configured.

Configuring network parameters in a DHCPv6 option group

About this task

A DHCPv6 option group can be created by using the following methods:

- Create a static DHCPv6 option group by using the **ipv6 dhcp option-group** command. The static DHCPv6 option group takes precedence over the dynamic DHCPv6 option group.
- When the device acts as a DHCPv6 client, it automatically creates a dynamic DHCPv6 option group for saving the obtained parameters. For more information about creating a dynamic DHCPv6 option group, see "Configuring the DHCPv6 client."

Procedure

1. Enter system view.

   system-view

2. Create a static DHCPv6 option group and enter its view.

   ipv6 dhcp option-group *option-group-number*

3. Specify a DNS server address.

   dns-server *ipv6-address*

   By default, no DNS server address is specified.

4. Specify a domain name suffix.

   domain-name *domain-name*

   By default, no domain name suffix is specified.

5. Specify a SIP server address or domain name.

   sip-server { **address** *ipv6-address* | **domain-name** *domain-name* }

   By default, no SIP server address or domain name is specified.

6. Configure a self-defined DHCPv6 option.

   option *code* **hex** *hex-string*

By default, no self-defined DHCPv6 option is configured.

7. Return to system view.

   quit

8. Enter DHCPv6 address pool view.

   ipv6 dhcp pool *pool-name*

9. Specify a DHCPv6 option group.

   option-group *option-group-number*

   By default, no DHCPv6 option group is specified.

Configuring the DHCPv6 server on an interface

About this task

Enable the DHCP server and configure one of the following address/prefix assignment methods on an interface:

- **Apply an address pool on the interface**—The DHCPv6 server selects an IPv6 address/prefix from the applied address pool for a requesting client. If there is no assignable IPv6 address/prefix in the address pool, the DHCPv6 server cannot to assign an IPv6 address/prefix to a client.
- **Configure global address assignment on the interface**—The DHCPv6 server selects an IPv6 address/prefix in the global DHCPv6 address pool that matches the server interface address or the DHCPv6 relay agent address for a requesting client.

If you configure both methods on an interface, the DHCPv6 server uses the specified address pool for address assignment without performing global address assignment.

Restrictions and guidelines

- An interface cannot act as a DHCPv6 server and DHCPv6 relay agent at the same time.
- Do not enable DHCPv6 server and DHCPv6 client on the same interface.
- You can apply an address pool that has not been created to an interface. The setting takes effect after the address pool is created.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the DHCPv6 server on the interface.

   ipv6 dhcp select server

   By default, the interface does not act as a DHCP server or a DHCP relay agent, and discards DHCPv6 packets from DHCPv6 clients.

4. Configure an assignment method.
   - Configure global address assignment.

     ipv6 dhcp server { **allow-hint** | **preference** *preference-value* | **rapid-commit** } *

     By default, the server supports global address assignment, but does not support desired address/prefix assignment or rapid assignment. The server preference is not set.
   - Apply a DHCPv6 address pool to the interface.

     ipv6 dhcp server apply pool *pool-name* [**allow-hint** | **preference** *preference-value* | **rapid-commit**] *

By default, no DHCPv6 address pool is applied to the interface.

Configuring a DHCPv6 policy for IPv6 address and prefix assignment

About this task

In a DHCPv6 policy, each DHCPv6 user class has a bound DHCPv6 address pool. Clients matching different user classes obtain IPv6 addresses, IPv6 prefixes, and other parameters from different address pools. When receiving a DHCPv6 request, the DHCPv6 server compares the packet against the user classes in the order that they are configured.

If a match is found and the bound address pool has assignable IPv6 addresses or prefixes, the server uses the address pool for assignment. If the bound address pool does not have assignable IPv6 addresses or prefixes, the assignment fails.

If no match is found, the server uses the default DHCPv6 address pool for assignment. If no default address pool is specified or the default address pool does not have assignable IPv6 addresses or prefixes, the assignment fails.

For successful assignment, make sure the applied DHCPv6 policy and the bound address pools exist.

A match rule cannot match an option added by the DHCPv6 device, for example, Option 18 or Option 37.

Procedure

1. Enter system view.

   system-view

2. Create a DHCPv6 user class and enter DHCPv6 user class view.

   ipv6 dhcp class *class-name*

3. Configure a match rule for the DHCPv6 user class.

   if-match rule *rule-number* { **option** *option-code* [**ascii** *ascii-string* [**offset** *offset* | **partial**] | **hex** *hex-string* [**mask** *mask* | **offset** *offset* **length** *length* | **partial**]] | **relay-agent** *gateway-ipv6-address* }

   By default, no match rule is configured for a DHCPv6 user class.

4. Return to system view.

   quit

5. Create a DHCPv6 policy and enter DHCPv6 policy view.

   ipv6 dhcp policy *policy-name*

   The DHCPv6 policy takes effect only after it is applied to the interface that acts as the DHCPv6 server.

6. Specify a DHCPv6 address pool for a DHCPv6 user class.

   class *class-name* **pool** *pool-name*

   By default, no address pool is specified for a user class.

7. (Optional.) Specify the default DHCPv6 address pool.

   default pool *pool-name*

   By default, the default address pool is not specified.

8. Return to system view.

   quit

9. Enter interface view.

```
interface interface-type interface-number
```

10. Apply the DHCPv6 policy to the interface.

```
ipv6 dhcp apply-policy policy-name
```

By default, no DHCPv6 policy is applied to an interface.

Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 server

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.

```
system-view
```

2. Set the DSCP value for DHCPv6 packets sent by the DHCPv6 server.

```
ipv6 dhcp dscp dscp-value
```

By default, the DSCP value in DHCPv6 packets sent by the DHCPv6 server is 56.

Configuring DHCPv6 binding auto backup

About this task

The auto backup feature saves DHCPv6 bindings to a backup file, and allows the DHCPv6 server to download the bindings from the backup file at the server reboot. The bindings include the lease bindings and conflicted IPv6 addresses. They cannot survive a reboot on the DHCPv6 server.

The DHCPv6 server does not provide services during the download process. If a connection error occurs during the process and cannot be repaired in a short amount of time, you can terminate the download operation. Manual interruption allows the DHCPv6 server to provide services without waiting for the connection to be repaired.

Procedure

1. Enter system view.

```
system-view
```

2. Configure the DHCPv6 server to back up the bindings to a file.

```
ipv6 dhcp server database filename { filename | url url [ username
username [ password { cipher | simple } string ] ] }
```

By default, the DHCPv6 server does not back up the DHCPv6 bindings.

With this command executed, the DHCPv6 server backs up its bindings immediately and runs auto backup.

3. (Optional.) Manually save the DHCPv6 bindings to the backup file.

```
ipv6 dhcp server database update now
```

4. (Optional.) Set the waiting time after a DHCPv6 binding change for the DHCPv6 server to update the backup file.

```
ipv6 dhcp server database update interval interval
```

By default, the DHCP server waits 300 seconds to update the backup file after a DHCP binding change. If no DHCP binding changes, the backup file is not updated.

5. (Optional.) Terminate the download of DHCPv6 bindings from the backup file.

ipv6 dhcp server database update stop

This command only triggers one termination.

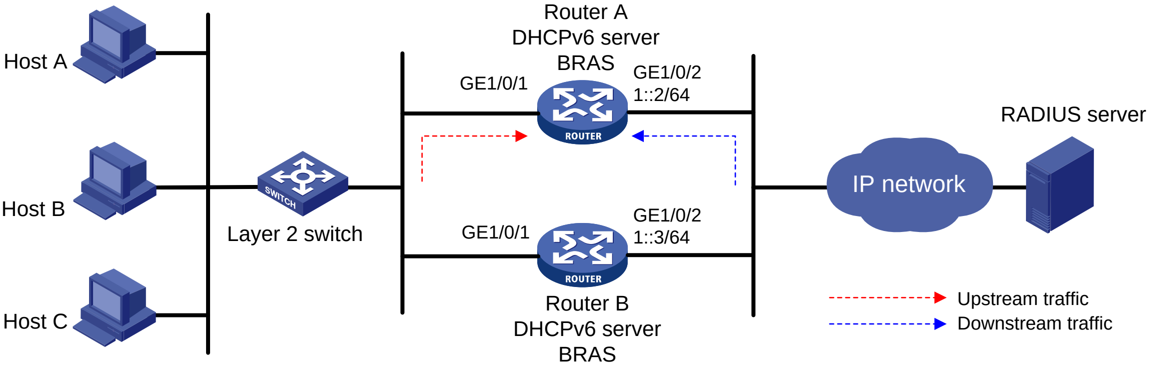
Advertising subnets assigned to clients

About this task

This feature enables the route management module to advertise subnets assigned to DHCPv6 clients. This feature achieves symmetric routing for traffic of the same host.

As shown in Figure 11, Router A and Router B act as both the DHCPv6 server and the BRAS device. The BRAS devices send accounting packets to the RADIUS server. To enable the BRAS devices to collect correct accounting information for each RADIUS user, configure the DHCPv6 server to advertise subnets assigned to clients. The upstream and downstream traffic of a RADIUS user will pass through the same BRAS device.

Figure 11 Network diagram



Procedure

1. Enter system view.

   system-view

2. Enter DHCP address pool view.

   ipv6 dhcp pool *pool-name*

3. Advertise the subnet that is assigned to DHCPv6 clients.

   network { *prefix/prefix-length* | **prefix** *prefix-number* [*sub-prefix/sub-prefix-length*] } [**preferred-lifetime** *preferred-lifetime* **valid-lifetime** *valid-lifetime*] **export-route**

   By default, the subnet assigned to DHCPv6 clients is not advertised.

Applying a DHCPv6 address pool to a VPN instance

About this task

If a DHCPv6 address pool is applied to a VPN instance, the DHCPv6 server assigns IPv6 addresses in this address pool to clients in the VPN instance. Addresses in this address pool will not be assigned to clients on the public network.

The DHCPv6 server can obtain the VPN instance to which a DHCPv6 client belongs from the following information:

- The client's VPN information stored in authentication modules.
- The VPN information of the DHCPv6 server's interface that receives DHCPv6 packets from the client.

The VPN information from authentication modules takes priority over the VPN information of the receiving interface.

Procedure

1. Enter system view.
   `system-view`
2. Enter DHCP address pool view.
   `ipv6 dhcp pool` *`pool-name`*
3. Apply the address pool to a VPN instance.
   `vpn-instance` *`vpn-instance-name`*
   By default, the address pool is not applied to any VPN instance.

Enabling DHCPv6 logging on the DHCPv6 server

About this task

The DHCPv6 logging feature enables the DHCPv6 server to generate DHCPv6 logs and send them to the information center. The information helps administrators locate and solve problems. For information about the log destination and output rule configuration in the information center, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

As a best practice, disable this feature if the log generation affects the device performance or reduces the address and prefix allocation efficiency. For example, this situation might occur when a large number of clients frequently come online or go offline.

Procedure

1. Enter system view.
   `system-view`
2. Enable DHCPv6 logging.
   `ipv6 dhcp log enable`
   By default, DHCPv6 logging is disabled.

Display and maintenance commands for DHCPv6 server

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display the DUID of the local device.	**`display ipv6 dhcp duid`**
Display information about a DHCPv6 option group.	**`display ipv6 dhcp option-group`** [*`option-group-number`*]
Display DHCPv6 address pool information.	**`display ipv6 dhcp pool`** [*`pool-name`* | **`vpn-instance`** *`vpn-instance-name`*]

Task	Command
Display prefix pool information.	**display ipv6 dhcp prefix-pool** [*prefix-pool-number*] [**vpn-instance** *vpn-instance-name*]
Display DHCPv6 server information on an interface.	**display ipv6 dhcp server** [**interface** *interface-type interface-number*]
Display information about IPv6 address conflicts.	**display ipv6 dhcp server conflict** [**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*]
Display information about DHCPv6 binding auto backup	**display ipv6 dhcp server database**
Display information about expired IPv6 addresses.	**display ipv6 dhcp server expired** [[**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Display information about IPv6 address bindings.	**display ipv6 dhcp server ip-in-use** [[**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Display information about IPv6 prefix bindings.	**display ipv6 dhcp server pd-in-use** [**pool** *pool-name* | [**prefix** *prefix/prefix-len*] [**vpn-instance** *vpn-instance-name*]]
Display packet statistics on the DHCPv6 server.	**display ipv6 dhcp server statistics** [**pool** *pool-name* | **vpn-instance** *vpn-instance-name*]
Clear information about IPv6 address conflicts.	**reset ipv6 dhcp server conflict** [**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*]
Clear information about expired IPv6 address bindings.	**reset ipv6 dhcp server expired** [[**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Clear information about IPv6 address bindings.	**reset ipv6 dhcp server ip-in-use** [[**address** *ipv6-address*] [**vpn-instance** *vpn-instance-name*] | **pool** *pool-name*]
Clear information about IPv6 prefix bindings.	**reset ipv6 dhcp server pd-in-use** [**pool** *pool-name* | [**prefix** *prefix/prefix-len*] [**vpn-instance** *vpn-instance-name*]]
Clear packets statistics on the DHCPv6 server.	**reset ipv6 dhcp server statistics** [**vpn-instance** *vpn-instance-name*]

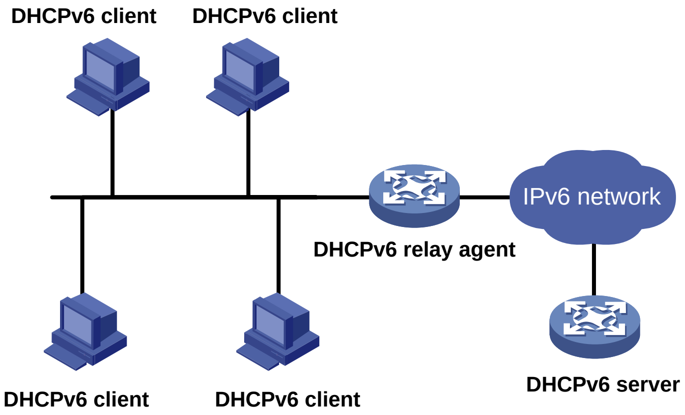
Configuring the DHCPv6 relay agent

About DHCPv6 relay agent

Typical application

A DHCPv6 client usually uses a multicast address to contact the DHCPv6 server on the local link to obtain an IPv6 address and other configuration parameters. As shown in Figure 12, if the DHCPv6 server resides on another subnet, the DHCPv6 clients need a DHCPv6 relay agent to contact the server. The relay agent feature avoids deploying a DHCPv6 server on each subnet.

Figure 12 Typical DHCPv6 relay agent application

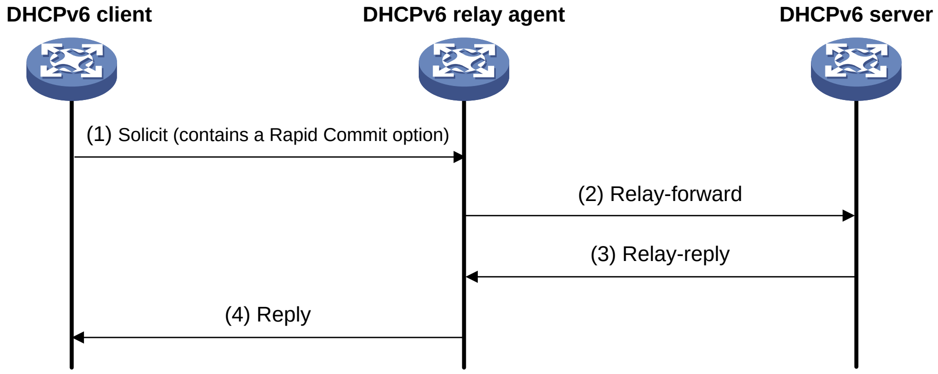


DHCPv6 relay agent operating process

As shown in Figure 13, a DHCPv6 client obtains an IPv6 address and other network configuration parameters from a DHCPv6 server through a DHCPv6 relay agent. The following example uses rapid assignment to describe the process:

- The DHCPv6 client sends a Solicit message containing the Rapid Commit option to the multicast address FF02::1:2 of all the DHCPv6 servers and relay agents.
- After receiving the Solicit message, the DHCPv6 relay agent encapsulates the message into the Relay Message option of a Relay-forward message, and sends the message to the DHCPv6 server.
- After obtaining the Solicit message from the Relay-forward message, the DHCPv6 server performs the following tasks:
  - Selects an IPv6 address and other required parameters.
  - Adds them to a reply that is encapsulated within the Relay Message option of a Relay-reply message.
  - Sends the Relay-reply message to the DHCPv6 relay agent.
- The DHCPv6 relay agent obtains the reply from the Relay-reply message and sends the reply to the DHCPv6 client.
- The DHCPv6 client uses the IPv6 address and other network parameters assigned by the DHCPv6 server to complete network configuration.

Figure 13 Operating process of a DHCPv6 relay agent



DHCPv6 relay agent tasks at a glance

To configure a DHCPv6 relay agent, perform the following tasks:

1. Enabling the DHCPv6 relay agent on an interface
2. Specifying DHCPv6 servers on the relay agent
3. (Optional.) Specifying a gateway address for DHCPv6 clients
4. (Optional.) Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 relay agent
5. (Optional.) Specifying a padding mode for the Interface-ID option

Enabling the DHCPv6 relay agent on an interface

Restrictions and guidelines

As a best practice, do not enable DHCPv6 relay agent and DHCPv6 client on the same interface.

Procedure

1. Enter system view.
   `system-view`
2. Enter interface view.
   `interface` *`interface-type interface-number`*
3. Enable DHCPv6 relay agent on the interface.
   `ipv6 dhcp select relay`
   By default, the DHCPv6 relay agent is disabled on the interface.

Specifying DHCPv6 servers on the relay agent

Specifying DHCPv6 server IP addresses

Restrictions and guidelines

- You can use the **`ipv6 dhcp relay server-address`** command to specify a maximum of eight DHCPv6 servers on the DHCPv6 relay agent interface. The DHCPv6 relay agent forwards DHCP requests to all the specified DHCPv6 servers.

- If a DHCPv6 server address is a link-local address or multicast address, you must specify an outgoing interface by using the **`interface`** keyword in this command. Otherwise, DHCPv6 packets might fail to reach the DHCPv6 server.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify a DHCPv6 server.

   `ipv6 dhcp relay server-address` *`ipv6-address`* [**`interface`** *`interface-type interface-number`*]

   By default, no DHCPv6 server is specified.

Specifying DHCPv6 servers for a DHCPv6 address pool on the DHCPv6 relay agent

About this task

This feature allows DHCPv6 clients of the same type to obtain IPv6 addresses, IPv6 prefixes, and other configuration parameters from the DHCPv6 servers in the matching DHCPv6 address pool.

It applies to scenarios where the DHCPv6 relay agent connects to clients of the same access type but classified into different types by their locations. In this case, the relay interface typically has no IPv6 address configured. You can use the **`gateway-list`** command to specify the gateway addresses for clients matching the same DHCPv6 address pool.

Upon receiving a DHCPv6 Solicit or Request from a client that matches a DHCPv6 address pool, the relay agent processes the packet as follows:

- Fills the **link-address** field of the packet with a specified gateway address.
- Forwards the packet to all DHCPv6 servers in the matching DHCPv6 address pool.

The DHCPv6 servers select a DHCPv6 address pool according to the gateway address.

Restrictions and guidelines

- You can specify a maximum of eight DHCPv6 servers for one DHCPv6 address pool for high availability. The relay agent forwards DHCPv6 Solicit and Request packets to all DHCPv6 servers in the DHCPv6 address pool.
- If this feature is used in the PPPoE scenario, execute the **`ipv6 dhcp relay client-information record`** command to enable the DHCPv6 relay agent to record relay entries. When a PPPoE user gets offline, the DHCPv6 relay agent locates the matching relay entry and sends a Release message to the DHCPv6 server.
- If this feature is used in the PPPoE scenario, you do not need to execute the **`ipv6 dhcp select relay`** command. This is because the **`remote-server`** command is a must in this configuration task and it implies that this device is a relay device.

Procedure

1. Enter system view.

   `system-view`

2. Create a DHCPv6 address pool and enter its view.

   `ipv6 dhcp pool` *`pool-name`*

3. Specify gateway addresses for the clients matching the DHCPv6 address pool.

```
gateway-list ipv6-address&<1-8>
```

By default, no gateway address is specified.

4. Specify DHCPv6 servers for the DHCPv6 address pool.

   ```
   remote-server ipv6-address [ interface interface-type
   interface-number ]
   ```

   By default, no DHCPv6 server is specified for the DHCPv6 address pool.

Specifying a gateway address for DHCPv6 clients

About this task

By default, the DHCPv6 relay agent fills the **link-address** field of DHCPv6 Solicit and Request packets with the first IPv6 address of the relay interface. You can specify a gateway address on the relay agent for DHCPv6 clients. The DHCPv6 relay agent uses the specified gateway address to fill the **link-address** field of DHCPv6 Solicit and Request packets.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify a gateway address for DHCPv6 clients.

   ```
   ipv6 dhcp relay gateway ipv6-address
   ```

   By default, the DHCPv6 relay agent uses the first IPv6 address of the relay interface as the clients' gateway address.

Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 relay agent

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the DSCP value for DHCPv6 packets sent by the DHCPv6 relay agent.

   ```
   ipv6 dhcp dscp dscp-value
   ```

   The default DSCP value is 56.

Specifying a padding mode for the Interface-ID option

About this task

This feature enables the relay agent to fill the Interface-ID option in the specified mode. When receiving a DHCPv6 packet from a client, the relay agent fills the Interface-ID option in the mode and then forwards the packet to the DHCPv6 server.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify a padding mode for the Interface-ID option.

   `ipv6 dhcp relay interface-id` { **`bas`** | **`interface`** }

   By default, the relay agent fills the Interface-ID option with the interface index of the interface.

Display and maintenance commands for DHCPv6 relay agent

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display the DUID of the local device.	**`display ipv6 dhcp duid`**
Display DHCPv6 server addresses specified on the DHCPv6 relay agent.	**`display ipv6 dhcp relay server-address`** [**`interface`** *`interface-type interface-number`*]
Display packet statistics on the DHCPv6 relay agent.	**`display ipv6 dhcp relay statistics`** [**`interface`** *`interface-type interface-number`*]
Clear packets statistics on the DHCPv6 relay agent.	**`reset ipv6 dhcp relay statistics`** [**`interface`** *`interface-type interface-number`*]

Configuring the DHCPv6 client

About the DHCPv6 client

With DHCPv6 client configured, an interface can obtain configuration parameters from the DHCPv6 server.

A DHCPv6 client can use DHCPv6 to complete the following functions:

- Obtain an IPv6 address, an IPv6 prefix, or both, and obtain other configuration parameters. If DHCPv6 server is enabled on the device, the client can automatically save the obtained parameters to a DHCPv6 option group. With the obtained IPv6 prefix, the client can generate its global unicast address.
- Support stateless DHCPv6 to obtain configuration parameters except IPv6 address and IPv6 prefix. The client obtains an IPv6 address through stateless IPv6 address autoconfiguration. If the client receives an RA message with the M flag set to 0 and the O flag set to 1 during address acquisition, stateless DHCPv6 starts.

Restrictions and guidelines: DHCPv6 client configuration

Do not configure the DHCPv6 client on the same interface as the DHCPv6 server or the DHCPv6 relay agent.

DHCPv6 client tasks at a glance

To configure a DHCPv6 client, perform the following tasks:

1. (Optional.) Configuring the DHCPv6 client DUID
2. Configuring the DHCPv6 client to obtain IPv6 addresses, IPv6 prefixes and other network parameters

   Choose the following tasks as needed:

   - Configuring IPv6 address acquisition
   - Configuring IPv6 prefix acquisition
   - Configuring IPv6 address and prefix acquisition
   - Configuring acquisition of configuration parameters except IP addresses and prefixes
3. (Optional.) Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 client

Configuring the DHCPv6 client DUID

About this task

The DUID of a DHCPv6 client is the globally unique identifier of the client. The client pads its DUID into Option 1 of the DHCPv6 packet that it sends to the DHCPv6 server. The DHCPv6 server can assign specific IPv6 addresses or prefixes to DHCPv6 clients with specific DUIDs.

Restrictions and guidelines

Make sure the DUID that you configure is unique.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the DHCPv6 client DUID.

   ipv6 dhcp client duid { **ascii** *ascii-string* | **hex** *hex-string* | **mac** *interface-type interface-number* }

   By default, the interface uses the device bridge MAC address to generate its DHCPv6 client DUID.

Configuring IPv6 address acquisition

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface to use DHCPv6 to obtain an IPv6 address and other configuration parameters.

   ipv6 address dhcp-alloc [**option-group** *group-number* | **rapid-commit**] *****

   By default, the interface does not use DHCPv6 for IPv6 address acquisition.

Configuring IPv6 prefix acquisition

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface to use DHCPv6 to obtain an IPv6 prefix and other configuration parameters.

   ipv6 dhcp client pd *prefix-number* [**option-group** *group-number* | **rapid-commit**] *

   By default, the interface does not use DHCPv6 for IPv6 prefix acquisition.

Configuring IPv6 address and prefix acquisition

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface to use DHCPv6 to obtain an IPv6 address, an IPv6 prefix, and other configuration parameters.

   ipv6 dhcp client stateful prefix *prefix-number* [**option-group** *option-group-number* | **rapid-commit**] *

   By default, the interface does not use DHCPv6 for IPv6 address and prefix acquisition.

Configuring acquisition of configuration parameters except IP addresses and prefixes

About this task

When a DHCPv6 client has obtained an IPv6 address and prefix, you can configure the following methods for the client to obtain other network configuration parameters:

- Execute the **`ipv6 address auto`** command to enable an interface to automatically generate an IPv6 global unicast address and a link-local address. Then stateless DHCPv6 will be triggered when the M flag is set to 0 and the O flag is set to 1 in a received RA message. For more information about the commands, see *Layer 3—IP services Command Reference*.
- Executing the **`ipv6 dhcp client stateless enable`** command on an interface to enable the interface to act as a DHCPv6 client to obtain configuration parameters from a DHCPv6 server.

If you execute both the **`ip address auto`** and **`ipv6 dhcp client stateless enable`** commands, the interface acts as follows:

- Generate a global unicast address and a link-local address.
- Obtain other configuration parameters from a DHCPv6 server.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enter interface view.
   ```
   interface interface-type interface-number
   ```
3. Configure the interface to support stateless DHCPv6. Choose the options to configure as needed:
   - Enable stateless IPv6 address autoconfiguration:
     ```
     ipv6 address auto
     ```
   - Configure the client to obtain network parameters from DHCPv6 servers:
     ```
     ipv6 dhcp client stateless enable
     ```

   By default, the interface does not support stateless DHCPv6.

Setting the DSCP value for DHCPv6 packets sent by the DHCPv6 client

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Set the DSCP value for DHCPv6 packets sent by the DHCPv6 client.
   ```
   ipv6 dhcp client dscp dscp-value
   ```
   By default, the DSCP value in DHCPv6 packets sent by the DHCPv6 client is 56.

Display and maintenance commands for DHCPv6 client

Execute the **display** commands in any view, and execute the **reset** command in user view.

Task	Command
Display the DHCPv6 client information.	**display ipv6 dhcp client** [**interface** *interface-type interface-number*]
Display the DHCPv6 client statistics.	**display ipv6 dhcp client statistics** [**interface** *interface-type interface-number*]
Clear the DHCPv6 client statistics.	**reset ipv6 dhcp client statistics** [**interface** *interface-type interface-number*]

Contents

Configuring DNS

About DNS

Domain Name System (DNS) is a distributed database used by TCP/IP applications to translate domain names into IP addresses. The domain name-to-IP address mapping is called a DNS entry.

Types of DNS services

DNS services can be static or dynamic. After a user specifies a name, the device checks the static name resolution table for an IP address. If no IP address is available, it uses a DNS server group or a DNS server for dynamic name resolution, which takes more time than static name resolution. To improve efficiency, you can put frequently queried name-to-IP address mappings in the local static name resolution table.

Static domain name resolution

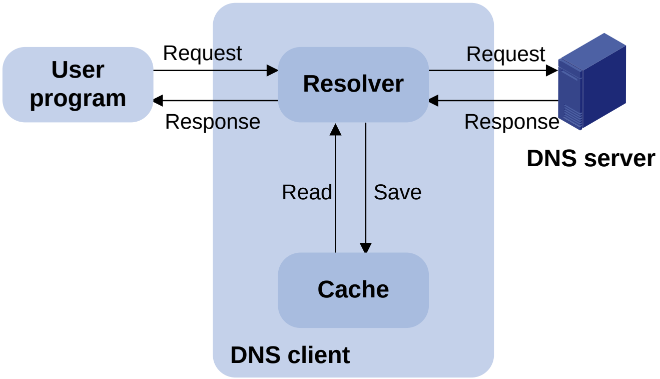
Static domain name resolution means manually creating mappings between domain names and IP addresses. For example, you can create a static DNS mapping for a device so that you can Telnet to the device by using the domain name.

DNS server-based domain name resolution

Architecture

Figure 1 shows the relationship between the user program, DNS client, and DNS server. The DNS client includes the resolver and cache. The user program and DNS client can run on the same device or different devices. The DNS server and the DNS client usually run on different devices.

Figure 1 DNS server-based dynamic domain name resolution



The device can function as a DNS client, but not a DNS server.

If an alias is configured for a domain name on the DNS server, the device can resolve the alias into the IP address of the host.

Resolution process

The DNS server-based dynamic domain name resolution process is as follows:

1. A user program sends a name query to the resolver of the DNS client.

2. The DNS resolver looks up the local domain name cache for a match. If the resolver finds a match, it sends the corresponding IP address back. If not, it sends a query to the DNS server.
3. The DNS server looks up the corresponding IP address of the domain name in its DNS database. If no match is found, the server sends a query to other DNS servers. This process continues until a result, whether successful or not, is returned.
4. After receiving a response from the DNS server, the DNS client returns the resolution result to the user program.

Caching

DNS server-based dynamic domain name resolution allows the DNS client to store latest DNS entries in the DNS cache. The DNS client does not need to send a request to the DNS server for a repeated query within the aging time. To make sure the entries from the DNS server are up to date, a DNS entry is removed when its aging timer expires. The DNS server determines how long a mapping is valid, and the DNS client obtains the aging information from DNS responses.

DNS suffixes

You can configure a domain name suffix list so that the resolver can use the list to supply the missing part of an incomplete name.

For example, you can configure **com** as the suffix for aabbcc.com. The user only needs to enter **aabbcc** to obtain the IP address of aabbcc.com. The resolver adds the suffix and delimiter before passing the name to the DNS server.

The name resolver handles the queries based on the domain names that the user enters:

- If the user enters a domain name without a dot (.) (for example, aabbcc), the resolver considers the domain name to be a host name. It adds a DNS suffix to the host name before performing the query operation. If no match is found for any host name and suffix combination, the resolver uses the user-entered domain name (for example, aabbcc) for the IP address query.
- If the user enters a domain name with a dot (.) among the letters (for example, www.aabbcc), the resolver directly uses this domain name for the query operation. If the query fails, the resolver adds a DNS suffix for another query operation.
- If the user enters a domain name with a dot (.) at the end (for example, aabbcc.com.), the resolver considers the domain name an FQDN and returns the successful or failed query result. The dot at the end of the domain name is considered a terminating symbol.

DNS proxy

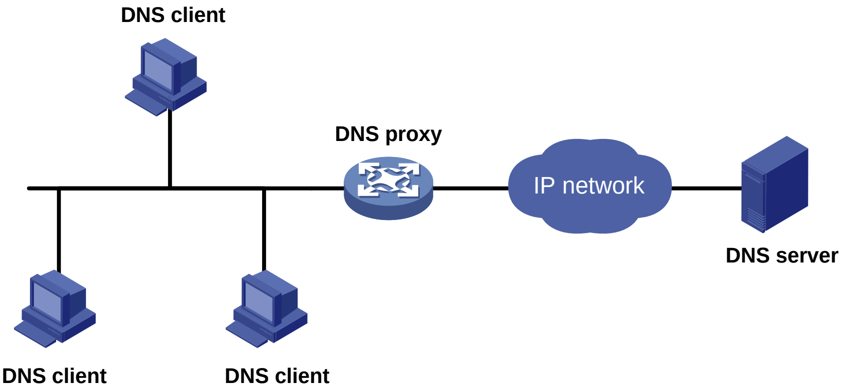
The DNS proxy performs the following functions:

- Forwards the request from the DNS client to the designated DNS server.
- Conveys the reply from the DNS server to the client.

The DNS proxy simplifies network management. When the DNS server address is changed, you can change the configuration only on the DNS proxy instead of on each DNS client.

Figure 2 shows the typical DNS proxy application.

Figure 2 DNS proxy application



A DNS proxy operates as follows:

1. A DNS client considers the DNS proxy as the DNS server, and sends a DNS request to the DNS proxy. The destination address of the request is the IP address of the DNS proxy.
2. The DNS proxy searches the local static domain name resolution table and dynamic domain name resolution cache after receiving the request. If the requested information is found, the DNS proxy returns a DNS reply to the client.
3. If the requested information is not found, the DNS proxy forwards the request as follows:
    a. If a matching domain name rule exists, the proxy forwards the request to DNS servers in the DNS server group bound to the rule.
    b. If no matching DNS server group exists but a DNS server is specified, the DNS proxy sends the request to the DNS server for domain name resolution.
4. After receiving a reply from the DNS server, the DNS proxy records the DNS mapping and forwards the reply to the DNS client.

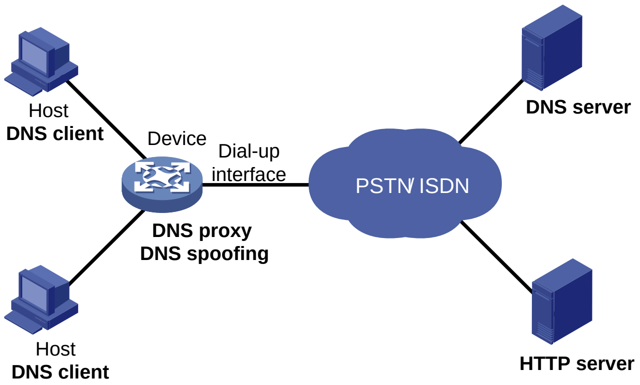
If no DNS server is designated or no route is available to the designated DNS server, the DNS proxy does not forward DNS requests.

DNS spoofing

As shown in Figure 3, DNS spoofing is applied to the dial-up network.

- The device connects to a PSTN network through a dial-up interface. The device triggers the establishment of a dial-up connection only when packets are to be forwarded through the dial-up interface.
- The device acts as a DNS proxy and is specified as a DNS server on the hosts. After the dial-up connection is established, the device dynamically obtains the DNS server address through DHCP or another autoconfiguration mechanism.

Figure 3 DNS spoofing application



The DNS proxy does not have the DNS server address or cannot reach the DNS server after startup. A host accesses the HTTP server in the following steps:

1. The host sends a DNS request to the device to resolve the domain name of the HTTP server into an IP address.
2. Upon receiving the request, the device searches the local static and dynamic DNS entries for a match. Because no match is found, the device spoofs the host by replying a configured IP address. The device must have a route to the IP address with the dial-up interface as the output interface.

   The IP address configured for DNS spoofing is not the actual IP address of the requested domain name. Therefore, the TTL field is set to 0 in the DNS reply. When the DNS client receives the reply, it creates a DNS entry and ages it out immediately.
3. Upon receiving the reply, the host sends an HTTP request to the replied IP address.
4. When forwarding the HTTP request through the dial-up interface, the device performs the following operations:
   - Establishes a dial-up connection with the network.
   - Dynamically obtains the DNS server address through DHCP or another autoconfiguration mechanism.
5. Because the DNS entry ages out immediately upon creation, the host sends another DNS request to the device to resolve the HTTP server domain name.
6. The device operates the same as a DNS proxy. For more information, see "DNS proxy."
7. After obtaining the IP address of the HTTP server, the host can access the HTTP server.

Without DNS spoofing, the device forwards the DNS requests from the host to the DNS server if it cannot find a matching local DNS entry. However, the device cannot obtain the DNS server address, because no dial-up connection is established. Therefore, the device cannot forward or answer the requests from the client. DNS resolution fails, and the client cannot access the HTTP server.

DNS tasks at a glance

To configure DNS, perform the following tasks:

1. Configuring the DNS client

   Choose the following tasks as needed:
   - Configuring static domain name resolution
   - Configuring DNS server-based domain name resolution
2. (Optional.) Configuring the DNS proxy
   - Configuring the DNS proxy

   - Configuring the DNS transparent proxy
3. (Optional.) Configuring DNS spoofing

   This feature is applied to the dial-up network.
4. (Optional.) Configuring DNS security features
   - Configuring DNS snooping
   - Enabling DNS snooping logging
   - Configuring a DNS packet rate limit
   - Configuring the DNS trusted interface
5. (Optional.) Configuring DNS packet parameters
   - Specifying the source interface for DNS packets
   - Setting the DSCP value for outgoing DNS packets
6. (Optional.) Configuring DNS filtering

Configuring the DNS client

About domain name resolution on the DNS client

A DNS client resolves a domain name in the following order:

1. Locally saved DNS mappings that have been resolved.
2. DNS server group-based domain name resolution.
3. Static domain name resolution.
4. DNS server-based domain name resolution.

The resolution fails if domain name cannot be resolved after all these methods are used.

Configuring static domain name resolution

Restrictions and guidelines

For the public network or a VPN instance, each host name maps to only one IPv4 address and one IPv6 address.

A maximum of 2048 DNS entries can be configured for the public network or each VPN instance. You can configure DNS entries for both public network and VPN instances.

Procedure

1. Enter system view.

   `system-view`
2. Configure a host name-to-address mapping.

   IPv4:

   `ip host` *`host-name ip-address`* [**`vpn-instance`** *`vpn-instance-name`*]

   IPv6:

   `ipv6 host` *`host-name ipv6-address`* [**`vpn-instance`** *`vpn-instance-name`*]

Configuring DNS server-based domain name resolution

Restrictions and guidelines

- The limit on the number of DNS servers on the device is as follows:

    - In system view, you can specify a maximum of six DNS server IPv4 addresses for the public network or each VPN instance. You can specify DNS server IPv4 addresses for both public network and VPN instances.
    - In system view, you can specify a maximum of six DNS server IPv6 addresses for the public network or each VPN instance. You can specify DNS server IPv6 addresses for both public network and VPN instances.
    - In interface view, you can specify a maximum of six DNS server IPv4 addresses for the public network or each VPN instance. You can specify DNS server IPv4 addresses for both public network and VPN instances.
- A DNS server address is required so that DNS queries can be sent to a correct server for resolution. If you specify both an IPv4 address and an IPv6 address, the device performs the following operations:
    - Sends an IPv4 DNS query first to the DNS server IPv4 addresses. If the query fails, the device turns to the DNS server IPv6 addresses.
    - Sends an IPv6 DNS query first to the DNS server IPv6 addresses. If the query fails, the device turns to the DNS server IPv4 addresses.
- A DNS server address specified in system view takes priority over a DNS server address specified in interface view. A DNS server address specified earlier has a higher priority. A DNS server address manually specified takes priority over a DNS server address dynamically obtained, for example, through DHCP. The device first sends a DNS query to the DNS server address of the highest priority. If the first query fails, it sends the DNS query to the DNS server address of the second highest priority, and so on.
- You can configure a DNS suffix that the system automatically adds to the incomplete domain name that a user enters.
    - You can configure a maximum of 16 DNS suffixes for the public network or each VPN instance. You can configure DNS suffixes for both public network and VPN instances.
    - A DNS suffix manually configured takes priority over a DNS suffix dynamically obtained, for example, through DHCP. A DNS suffix configured earlier has a higher priority. The device first uses the suffix that has the highest priority. If the query fails, the device uses the suffix that has the second highest priority, and so on.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the TTL value for DNS entries.

   dns cache ttl { **maximum** *max-value* | **minimum** *min-value* } *

   By default, the TTL value for DNS entries is the TTL value in the DNS reply.

3. (Optional.) Configure a DNS suffix.

   dns domain *domain-name* [**vpn-instance** *vpn-instance-name*]

   By default, no DNS suffix is configured and only the domain name that a user enters is resolved.

4. Specify a DNS server address.
    - Specify a DNS server address in system view.

      IPv4:

      dns server *ip-address* [**vpn-instance** *vpn-instance-name*]

      IPv6:

      ipv6 dns server *ipv6-address* [*interface-type interface-number*] [**vpn-instance** *vpn-instance-name*]
    - Execute the following commands in sequence to specify a DNS server IPv4 address in interface view.

      interface *interface-type interface-number*

dns server ip-address [**vpn-instance** *vpn-instance-name*]

By default, no DNS server address is specified.

Configuring the DNS proxy

Enabling DNS proxy

1. Enter system view.

   system-view

2. Enable DNS proxy.

   dns proxy enable

   By default, DNS proxy is disabled.

Specifying DNS server addresses

Restrictions and guidelines

You can specify multiple DNS servers. The DNS proxy forwards a request to the DNS server that has the highest priority. If it does not receive a reply, the proxy forwards the request to a DNS server that has the second highest priority, and so on.

You can specify both an IPv4 address and an IPv6 address.

- A DNS proxy forwards an IPv4 name query first to IPv4 DNS servers. If no reply is received, it forwards the request to IPv6 DNS servers.
- A DNS proxy forwards an IPv6 name query first to IPv6 DNS servers. If no reply is received, it forwards the request to IPv4 DNS servers.

Procedure

1. Enter system view.

   system-view

2. Specify a DNS server address.

   - Specify a DNS server address in system view.

     IPv4:

     dns server *ip-address* [**vpn-instance** *vpn-instance-name*]

     IPv6:

     ipv6 dns server *ipv6-address* [*interface-type interface-number*] [**vpn-instance** *vpn-instance-name*]

   - Execute the following commands in sequence to specify a DNS server IPv4 address in interface view.

     interface *interface-type interface-number*

     dns server ip-address [**vpn-instance** *vpn-instance-name*]

   By default, no DNS server address is specified.

Configuring the DNS transparent proxy

About this task

DNS transparent proxy modifies the source address in the DNS request so that the DNS client seems to receive a DNS reply directly from the DNS server. This feature is applicable to domain name-based policies, such as security policies and bandwidth policies.

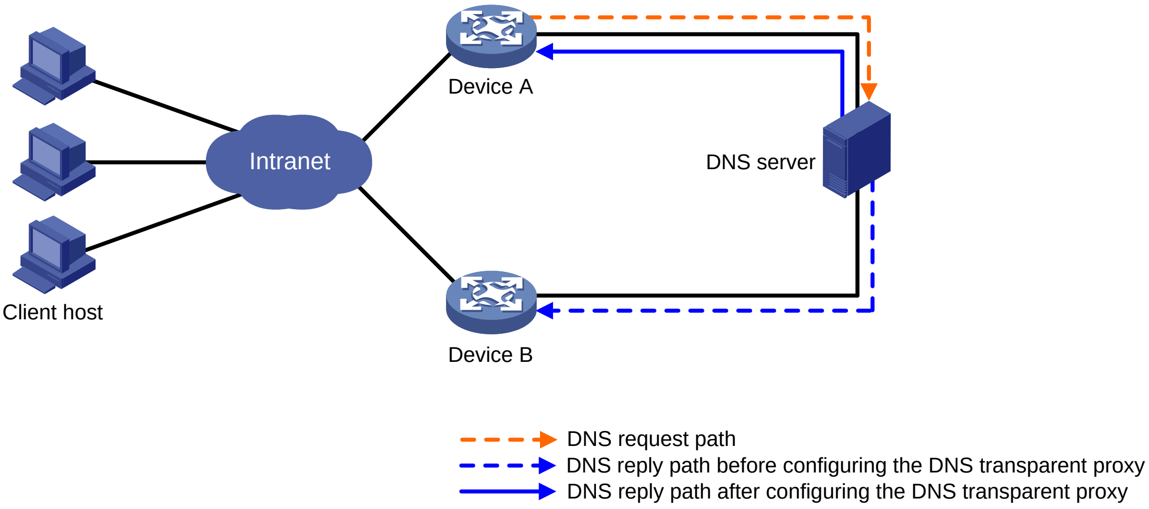
The DNS client does not configure the DNS server address as the DNS transparent proxy address, which simplifies DNS client configurations. As a best practice, enable DNS transparent proxy in some load balancing scenarios.

The device enabled with DNS transparent proxy monitors received DNS requests and replies and records the DNS mapping as follows:

1. The DNS transparent proxy monitors all received DNS packets. After receiving a DNS request, the DNS transparent proxy specifies a local IP address that can reach the DNS server as the source IP address in the request.
2. After receiving the DNS reply, the DNS transparent proxy records the DNS mapping and forwards the reply to the DNS client.
3. The DNS transparent proxy searches the local entries after receiving another request. If the requested information is found, the DNS transparent proxy returns a DNS reply to the client. If the requested information is not found, the DNS proxy forwards the query to the DNS server for domain name resolution.

Figure 4 shows the DNS transparent proxy conceptual diagram.

Figure 4 DNS transparent proxy conceptual diagram


Device A
DNS server
Intranet
Client host
Device B
DNS request path
DNS reply path before configuring the DNS transparent proxy
DNS reply path after configuring the DNS transparent proxy

A DNS transparency proxy operates as follows:

1. Device A is enabled with DNS transparency proxy. After receiving a DNS request, Device A changes the source address in the request to its own address and forwards the query to the DNS server.
2. After receiving the DNS reply, Device A records the DNS mapping and forwards the reply to the DNS client.

Restrictions and guidelines

The DNS transparent proxy and DNS snooping features cannot be both configured.

The DNS transparent proxy is not VPN-aware. The input interface and output interface of DNS packets must belong to the same VPN.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Set the TTL value for DNS entries.

   dns cache ttl { **maximum** *max-value* | **minimum** *min-value* } *

   By default, the TTL value for DNS entries is the TTL value in the DNS reply.

3. Enable DNS transparent proxy.

   dns transparent-proxy enable

   By default, DNS transparent proxy is disabled.

Configuring DNS spoofing

Restrictions and guidelines

- You can configure only one replied IPv4 address and one replied IPv6 address for the public network or a VPN instance. If you execute the command multiple times, the most recent configuration takes effect.
- You can configure DNS spoofing for both public network and VPN instances.
- After DNS spoofing takes effect, the device spoofs a DNS request even though a matching static DNS entry exists.

Prerequisites

The DNS proxy is enabled on the device.

No DNS server or route to any DNS server is specified on the device.

Procedure

1. Enter system view.

   system-view

2. Enable DNS proxy.

   dns proxy enable

   By default, DNS proxy is disabled.

3. Enable DNS spoofing and specify the IP address used to spoof DNS requests. Choose one option as needed:

   IPv4:

   dns spoofing *ip-address* [**vpn-instance** *vpn-instance-name*]

   IPv6:

   ipv6 dns spoofing *ipv6-address* [**vpn-instance** *vpn-instance-name*]

   By default, DNS spoofing is disabled.

Configuring DNS snooping

About this task

DNS snooping is applicable to domain name-based policies, such as security policies and bandwidth policies. To filter traffic based on domain names, the DNS mapping is required. The device enabled with DNS snooping monitors received DNS requests and replies. If the domain name in a DNS request matches a policy, the device records the DNS mapping after receiving a DNS reply and reports the mapping to the policy for traffic filtering. If the domain name does not match a policy, the device does not record the DNS mapping.

Restrictions and guidelines

DNS snooping only works between the DNS client and DNS server, or the DNS client and DNS proxy.

The DNS snooping and DNS transparent proxy features cannot be both configured.

The DNS snooping feature is not VPN-aware. The input interface and output interface of DNS packets must belong to the same VPN.

Procedure

1. Enter system view.

   `system-view`

2. (Optional.) Set the TTL value for DNS entries.

   dns cache ttl { **maximum** *max-value* | **minimum** *min-value* } *

   By default, the TTL value for DNS entries is the TTL value in the DNS reply.

3. Enable DNS snooping.

   `dns snooping enable`

   By default, DNS snooping is disabled.

Enabling DNS snooping logging

About this task

The DNS proxy searches the static domain name resolution table and dynamic domain name resolution cache after receiving a request.

- If the requested information is found, the DNS proxy returns a DNS reply to the client.
- If the requested information is not found, the DNS proxy sends the request to the designated DNS server.

Too many requests received at the same time will increase network load and affect the performance of the DNS proxy and DNS server. To avoid this issue, you can configure DNS snooping logging on the device between the DNS client and DNS proxy, or the DNS client and DNS server.

The device configured with DNS snooping monitors and records for received DNS queries and responses. Also, you can configure the device to generate and send DNS snooping logs to the fast log module. The administrator can locate and troubleshoot issues based on the logs. For information about the fast log output configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable DNS snooping logging.

   `dns snooping log enable`

   By default, DNS snooping logging is enabled.

Configuring a DNS packet rate limit

About this task

Perform this task to limit the rate of incoming DNS packets on interfaces. An interface will discard DNS packets exceeding the specified rate limit.

Restrictions and guidelines

This feature takes effect only when the DNS transparent proxy or DNS snooping logging feature is enabled.

Procedure

1. Enter system view.

   system-view

2. Set the DNS packet rate limit.

   dns snooping rate-limit *rate*

   By default, the rate of incoming DNS packets is not limited.

Configuring the DNS trusted interface

About this task

This task enables the device to use only the DNS suffix and domain name server information obtained through the trusted interface. The device can then obtain the correct resolved IP address. This feature protects the device against attackers that act as the DHCP server to assign incorrect DNS suffix and domain name server address.

Restrictions and guidelines

You can configure a maximum of 128 DNS trusted interfaces.

Procedure

1. Enter system view.

   system-view

2. Specify the DNS trusted interface.

   dns trust-interface *interface-type interface-number*

   By default, no DNS trusted interface is specified.

Specifying the source interface for DNS packets

About this task

This task enables the device to always use the primary IP address of the specified source interface as the source IP address of outgoing DNS packets. This feature applies to scenarios in which the DNS server responds only to DNS requests sourced from a specific IP address. If no IP address is configured on the source interface, no DNS packets can be sent out.

Restrictions and guidelines

When sending an IPv6 DNS request, the device follows the method defined in RFC 3484 to select an IPv6 address of the source interface.

You can configure only one source interface on the public network or a VPN instance. You can configure source interfaces for both public network and VPN instances.

Make sure the source interface belongs to the specified VPN instance if you specify the **vpn-instance** *vpn-instance-name* option.

Procedure

1. Enter system view.

   system-view

2. Specify the source interface for DNS packets.

dns source-interface *interface-type interface-number* [**vpn-instance** *vpn-instance-name*]

By default, no source interface for DNS packets is specified.

Setting the DSCP value for outgoing DNS packets

About this task

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet. A bigger DSCP value represents a higher priority.

Procedure

1. Enter system view.

   system-view

2. Set the DSCP value for DNS packets sent by a DNS client or a DNS proxy.

   IPv4:

   dns dscp *dscp-value*

   By default, the DSCP value is 0 in IPv4 DNS packets sent by a DNS client or a DNS proxy.

   IPv6:

   ipv6 dns dscp *dscp-value*

   By default, the DSCP value is 0 in IPv6 DNS packets sent by a DNS client or a DNS proxy.

Configuring DNS filtering

About this task

Enabled with DNS filtering, the DNS proxy matches the domain names in DNS requests with the host names on the allowlist or denylist to filter in or discard DNS requests.

The DNS proxy uses DNS filtering to filter DNS requests as follows:

- If the allowlist has a matching host name or the denylist has no matching host name with the domain name in the received DNS request, the DNS proxy filters in the request. After receiving a DNS reply, the DNS proxy records the DNS mapping and forwards the reply to the DNS client.
- If the denylist has a matching host name or the allowlist has no matching host name with the domain name in the received DNS request, the DNS proxy discards the DNS request.

To implement a strict access control, use an allowlist to filter DNS requests. To implement a loose access control, use a denylist to filter DNS requests.

Restrictions and guidelines

To add multiple host names to the allowlist or denylist, repeat this command. However, a host name cannot be added to both the denylist and allowlist.

Procedure

1. Enter system view.

   system-view

2. Enable DNS filtering and add a host name to the allowlist or denylist.

   dns filter { **allowlist** | **denylist** } *hostname*

   By default, DNS filtering is disabled.

Display and maintenance commands for DNS

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display DNS suffixes.	**display dns domain** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Display the domain name resolution table.	**display dns host** [**ip** | **ipv6**] [**vpn-instance** *vpn-instance-name*]
Display IPv4 DNS server information.	**display dns server** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Display IPv6 DNS server information.	**display ipv6 dns server** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Clear dynamic DNS entries.	**reset dns host** [**ip** | **ipv6**] [**vpn-instance** *vpn-instance-name*]
Clear log statistics for incoming DNS packets.	**reset dns snooping log statistics**

Troubleshooting DNS configuration

Failure to resolve IPv4 addresses

Symptom

After enabling dynamic domain name resolution, the user cannot get the correct IP address.

Solution

To resolve the problem:

1. Use the **display dns host ip** command to verify that the specified domain name is in the cache.
2. If the specified domain name does not exist, check that the DNS client can communicate with the DNS server.
3. If the specified domain name is in the cache, but the IP address is incorrect, check that the DNS client has the correct IP address of the DNS server.
4. Verify that the mapping between the domain name and IP address is correct on the DNS server.

Failure to resolve IPv6 addresses

Symptom

After enabling dynamic domain name resolution, the user cannot get the correct IPv6 address.

Solution

To resolve the problem:

1. Use the **display dns host ipv6** command to verify that the specified domain name is in the cache.

2. If the specified domain name does not exist, check that dynamic domain name resolution is enabled, and that the DNS client can communicate with the DNS server.
3. If the specified domain name is in the cache, but the IPv6 address is incorrect, check that the DNS client has the correct IPv6 address of the DNS server.
4. Verify that the mapping between the domain name and IPv6 address is correct on the DNS server.

Contents

Optimizing IP performance

IP performance optimization tasks at a glance

All IP performance optimization tasks are optional.

1. Configuring features for IP packets
   - Enabling an interface to receive and forward directed broadcasts destined for the directly connected network
   - Setting the interface MTU for IPv4 packets
   - Enabling IPv4 local fragment reassembly

     This feature is applicable in IRF networks.
   - Enabling IPv4 virtual fragment reassembly
   - Enabling fragment centralization for IPv4 VFR
   - Forcibly disabling IPv4 VFR
   - Enabling fragment centralization for IPv6 VFR
   - Forcibly disabling IPv6 VFR
   - Configuring the DF bit for IP packets
2. Configuring features for ICMP messages
   - Enabling sending ICMP error messages
   - Configuring rate limit for ICMP error messages
   - Specifying the source address for ICMP packets
3. Configuring features for TCP packets
   - Setting TCP MSS for an interface
   - Configuring TCP path MTU discovery
   - Enabling SYN Cookie
   - Setting the TCP buffer size
   - Setting TCP timers
   - Enabling carrying the TCP timestamp option in outgoing TCP packets

Enabling an interface to receive and forward directed broadcasts destined for the directly connected network

About forwarding broadcasts destined for the directly connected network

A directed broadcast packet is destined for all hosts on a specific network. In the destination IP address of the directed broadcast, the network ID identifies the target network, and the host ID is made up of all ones.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the interface to receive and forward directed broadcasts destined for the directly connected network.

   ip forward-broadcast

   By default, an interface can receive directed broadcasts destined for the directly connected network, but it cannot forward these broadcasts.

Setting the interface MTU for IPv4 packets

About this task

The interface MTU for IPv4 packets defines the largest size of an IPv4 packet that an interface can transmit without fragmentation. When a packet exceeds the MTU of the sending interface, the device processes the packet in one of the following ways:

- If the packet disallows fragmentation, the device discards it.
- If the packet allows fragmentation, the device fragments it and forwards the fragments.

Fragmentation and reassembling consume system resources, so set the MTU based on the network environment to avoid fragmentation.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the interface MTU for IPv4 packets.

   ip mtu *mtu-size*

   By default, the interface MTU is not set.

Enabling IPv4 local fragment reassembly

About this task

Use this feature on a device to improve fragment reassembly efficiency. This feature enables the LPU to reassemble the IPv4 fragments of a packet if all the fragments arrive at it. If this feature is disabled, all IPv4 fragments are delivered to the active MPU for reassembly. The feature applies only to fragments destined for the same LPU.

Use this feature on a multichassis IRF fabric to improve fragment reassembly efficiency. This feature enables a subordinate to reassemble the IPv4 fragments of a packet if all the fragments arrive at it. If this feature is disabled, all IPv4 fragments are delivered to the master device for reassembly. The feature applies only to fragments destined for the same subordinate.

Procedure

1. Enter system view.

   system-view

2. Enable IPv4 local fragment reassembly.

   ```
   ip reassemble local enable
   ```

   By default, IPv4 local fragment reassembly is disabled.

Enabling IPv4 virtual fragment reassembly

About this task

To prevent each service module from processing packet fragments that do not arrive in order, you can enable the virtual fragment reassembly (VFR) feature. This feature virtually reassembles the fragments of a datagram through fragment check, sequencing, and caching, ensuring fragments arrive at each service module in order.

VFR can detect and prevent the following types of attacks:

- **Tiny fragment attack**—The first fragment size is too small to hold the Layer 4 (such as TCP and UDP) header field, which is forced into the second fragment. VFR discards all tiny fragments.
- **Overlapping fragment attack**—Two consecutive incoming fragments are identical or overlap with each other. If an overlapping fragment is detected, VFR discards all fragments within a fragment chain.
- **Fragment flooding attack**—The maximum number of concurrent preassemblies or the number of fragments per datagram exceeds the upper limits. VFR discards subsequent fragments if the upper limit is reached.

Restrictions and guidelines

The enabling status of VFR can be managed at CLI or the enabling status of a service module that can call VFR. VRF is enabled in either of the following conditions:

- A service module that can call it is enabled.
- The **ip virtual-reassembly enable** command is executed.

If fragment reassembly is required, but a service module cannot call it, execute this command at CLI.

The **ip virtual-reassembly suppress** command can forcibly disable the VFR feature enabled through CLI or service calling.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable IPv4 virtual fragment reassembly.

   ```
   ip virtual-reassembly enable
   ```

   By default, IPv4 virtual fragment reassembly is disabled.

Enabling fragment centralization for IPv4 VFR

About this task

On an HA network, if an HA device enabled with IPv4 VFR does not receive all fragments of a datagram, it cannot reassemble the datagram and will discard the received fragments. To resolve this issue, you can enable this feature. Devices that do not receive the first fragment of a datagram forward the received fragments of this datagram to the device that receives the first fragment for VFR.

Restrictions and guidelines

This feature is applicable to devices enabled with IPv4 VFR on a HA network.

For more information about HA network, see high availability group configuration in *High Availability Configuration Guide*.

Procedure

1. Enter system view.
   `system-view`
2. Enable fragment centralization for IPv4 VFR.
   `ip virtual-reassembly centralize`
   By default, fragment centralization is disabled for IPv4 VFR.

Forcibly disabling IPv4 VFR

About this task

IPv4 VFR checks, sequences, and caches fragments upon fragment receiving to ensure that fragments to assemble are in the correct order. By default, IPv4 VFR is enabled.

On an HA network, if an HA device does not receive all fragments of a datagram, it cannot reassemble the datagram and will discard the received fragments. For the devices to permit the received fragments to pass, you can forcibly disable IPv4 VFR. The **`ip virtual-reassembly suppress`** command can forcibly disable the VFR feature enabled through CLI or service calling.

With IPv4 VFR disabled forcibly, ASPF and connection limit do not take effect on the received IPv4 fragments and the fragments will be forwarded directly.

Restrictions and guidelines

Use this feature according to the demands of VFR.

For more information about HA network, see high availability group configuration in *High Availability Configuration Guide*.

Procedure

1. Enter system view.
   `system-view`
2. Forcibly disable IPv4 VFR.
   `ip virtual-reassembly suppress`
   By default, forcibly disabling IPv4 VFR is enabled.

Enabling fragment centralization for IPv6 VFR

About this task

On an HA network, if an HA device enabled with IPv6 VFR does not receive all fragments of a datagram, it cannot reassemble the datagram and will discard all the received fragments. To resolve this issue, you can enable this feature. Devices that do not receive the first fragment of a datagram forward the received fragments of this datagram to the device that receives the first fragment for VFR.

Restrictions and guidelines

This feature is applicable to devices enabled with IPv6 VFR on a HA network.

For more information about HA network, see high availability group configuration in *High Availability Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable fragment centralization for IPv6 VFR.

   `ipv6 virtual-reassembly centralize`

   By default, fragment centralization is disabled for IPv6 VFR.

Forcibly disabling IPv6 VFR

About this task

IPv6 VFR checks, sequences, and caches fragments upon fragment receiving to ensure that fragments to assemble are in the correct order. By default, IPv6 VFR is enabled.

On an HA network, if an HA device does not receive all fragments of a datagram, it cannot reassemble the datagram and will discard the received fragments. For the devices to permit the received fragments to pass, you can forcibly disable IPv6 VFR.

With IPv6 VFR disabled forcibly, ASPF and connection limit do not take effect on the received IPv6 fragments and the fragments will be forwarded directly.

Restrictions and guidelines

Use this feature according to the demands of VFR.

For more information about HA network, see high availability group configuration in *High Availability Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Forcibly disable IPv6 VFR.

   `ipv6 virtual-reassembly suppress`

   By default, forcibly disabling IPv6 VFR is enabled.

Configuring the DF bit for IP packets

About this task

This task configures the Don't Fragment (DF) bit for the IP packets to be forwarded:

- **set**—Sets the DF bit in the IP packets to 1 to prevent the devices on the path from fragmenting the IP packets. If the path MTU is less than the size of the IP packets and DF bit is set, communication interruption occurs. The devices on the path will drop the IP packets and reply ICMP error messages to the IP packet sender.
- **clear**—Sets the DF bit in the IP packets to 0. The devices can fragment the IP packets before forwarding them.

This feature does not apply to IP packets generated by the local device.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080,	Yes

NFNX3-HDB680, NFNX3-HDB1080	
NFNX3-HDB1180, NFNX3-HDB1480	No

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the DF bit for IP packets.

   ```
   ip df-bit { clear | set }
   ```

   By default, the DF bit value of IP packets is retained as it is.

Enabling sending ICMP error messages

About sending ICMP error messages

ICMP messages are used by network layer and transport layer protocols to communicate updates and errors with other devices, facilitating network management.

Sending excessive ICMP messages increases network traffic. The device performance degrades if it receives a lot of malicious ICMP messages that cause it to respond with ICMP error messages. To prevent such problems, the sending of ICMP error messages is disabled by default. You can enable sending ICMP error messages of different types as needed.

ICMP error messages include redirect messages, time exceeded messages, and destination unreachable messages.

Enabling sending ICMP redirect messages

About this task

A host that has only one default route sends all packets to the default gateway. The default gateway sends an ICMP redirect message to inform the host of a correct next hop when the following conditions are met:

- The receiving and sending interfaces are the same.
- The packet source IP address and the IP address of the packet receiving interface are on the same segment.
- There is no source route option in the received packet.

ICMP redirect messages simplify host management and enable hosts to gradually optimize their routing table.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable sending ICMP redirect messages.

   ```
   ip redirects enable
   ```

   By default, the sending of ICMP redirect messages is disabled.

Enabling sending ICMP time exceeded messages

About this task

A device sends ICMP time exceeded messages by following these rules:

- The device sends the source an ICMP TTL exceeded in transit message when the following conditions are met:
    - The received packet is not destined for the device.
    - The TTL field of the packet is 1.
- When the device receives the first fragment of an IP datagram destined for it, it starts a timer. If the timer expires before all the fragments of the datagram are received, the device sends an ICMP fragment reassembly time exceeded message to the source.

Restrictions and guidelines

If the ICMP time exceeded message sending is disabled, the device does not send ICMP TTL exceeded in transit messages. However, it can still send ICMP fragment reassembly time exceeded messages.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable sending ICMP time exceeded messages.

   ```
   ip ttl-expires enable
   ```

   By default, the sending of ICMP time exceeded messages is disabled.

Enable sending ICMP destination unreachable messages

About this task

A device sends ICMP destination unreachable messages by following these rules:

- The device sends the source an ICMP network unreachable message when the following conditions are met:
    - The packet does not match any route.
    - No default route exists in the routing table.
- The device sends the source an ICMP protocol unreachable message when the following conditions are met:
    - The packet is destined for the device.
    - The transport layer protocol of the packet is not supported by the device.
- The device sends the source an ICMP port unreachable message when the following conditions are met:
    - The UDP packet is destined for the device.
    - The packet's port number does not match the corresponding process.
- The device sends the source an ICMP source route failed message when the following conditions are met:
    - The source uses Strict Source Routing to send packets.
    - The intermediate device finds that the next hop specified by the source is not directly connected.
- The device sends the source an ICMP fragmentation needed and DF set message when the following conditions are met:
    - The MTU of the sending interface is smaller than the packet.

- The packet has DF set.

Restrictions and guidelines

If a DHCP-enabled device receives an ICMP echo reply without sending any ICMP echo requests, the device does not send any ICMP protocol unreachable messages to the source. To enable DHCP, use the **`dhcp enable`** command. For more information about this command, see *Layer 3—IP Services Command Reference*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable sending ICMP destination unreachable messages.

   ```
   ip unreachables enable
   ```

   By default, the sending of ICMP destination unreachable messages is disabled.

Configuring rate limit for ICMP error messages

About this task

To avoid sending excessive ICMP error messages within a short period that might cause network congestion, you can limit the rate at which ICMP error messages are sent. A token bucket algorithm is used with one token representing one ICMP error message.

A token is placed in the bucket at intervals until the maximum number of tokens that the bucket can hold is reached.

A token is removed from the bucket when an ICMP error message is sent. When the bucket is empty, ICMP error messages are not sent until a new token is placed in the bucket.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the interval for tokens to arrive in the bucket and the bucket size for ICMP error messages.

   ```
   ip icmp error-interval interval [ bucketsize ]
   ```

   By default, a token is placed in the bucket at intervals of 100 milliseconds and the bucket allows a maximum of 10 tokens.

   To disable the ICMP rate limit, set the interval to 0 milliseconds.

Specifying the source address for ICMP packets

About this task

Specifying the source IP address for outgoing ping echo requests and ICMP error messages helps users to locate the sending device easily. As a best practice, specify the IP address of the loopback interface as the source IP address.

Restrictions and guidelines

If you specify an IP address in the **`ping`** command, ping echo requests use the specified address as the source IP address rather than the IP address specified by the **`ip icmp source`** command.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Specify the source address for outgoing ICMP packets.

   ```
   ip icmp source [ vpn-instance vpn-instance-name ] ip-address
   ```

   By default, no source address is specified for outgoing ICMP packets. The default source IP addresses for different types of ICMP packets vary as follows:

   - For an ICMP error message, the source IP address is the IP address of the receiving interface of the packet that triggers the ICMP error message. ICMP error messages include Time Exceeded, Port Unreachable, and Parameter Problem messages.
   - For an ICMP echo request, the source IP address is the IP address of the sending interface.
   - For an ICMP echo reply, the source IP address is the destination IP address of the ICMP echo request specific to this reply.

Setting TCP MSS for an interface

About this task

The maximum segment size (MSS) option informs the receiver of the largest segment that the sender can accept. Each end announces its MSS during TCP connection establishment. If the size of a TCP segment is smaller than the MSS of the receiver, TCP sends the TCP segment without fragmentation. If not, it fragments the segment according to the receiver's MSS.

Restrictions and guidelines

- If you set the TCP MSS on an interface, the size of each TCP segment received or sent on the interface cannot exceed the MSS value.
- This configuration takes effect only for TCP connections established after the configuration rather than the TCP connections that already exist.
- This configuration is effective only for IP packets.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Set the TCP MSS for the interface.

   ```
   tcp mss value
   ```

   By default, the TCP MSS is not set.

Configuring TCP path MTU discovery

About this task

TCP path MTU discovery (in RFC 1191) discovers the path MTU between the source and destination ends of a TCP connection. The device uses the path MTU to calculate the MSS to avoid IP fragmentation. The path MTU uses an aging mechanism to ensure that the source device can increase the path MTU when the minimum link MTU on the path increases.

TCP path MTU discovery works as follows:

1. A TCP source device sends a packet with the Don't Fragment (DF) bit set.
2. A router discards the packet that exceeds the MTU of the outgoing interface and returns an ICMP error message. The error message contains the MTU of the outgoing interface.
3. Upon receiving the ICMP message, the TCP source device calculates the current path MTU of the TCP connection.

4. The TCP source device sends subsequent TCP segments that are smaller than the MSS (MSS = path MTU – IP header length – TCP header length).

If the TCP source device still receives ICMP error messages when the MSS is smaller than 32 bytes, the TCP source device will fragment packets.

An ICMP error message received from a router that does not support RFC 1191 has the MTU of the outgoing interface set to 0. Upon receiving the ICMP message, the TCP source device selects the path MTU smaller than the current path MTU from the MTU table as described in RFC 1191. Based on the selected path MTU, the TCP source device calculates the TCP MSS. The MTU table contains MTUs of 68, 296, 508, 1006, 1280, 1492, 2002, 4352, 8166, 17914, 32000, and 65535 bytes. Because the minimum TCP MSS specified by the system is 32 bytes, the actual minimum MTU is 72 bytes.

The aging mechanism of the path MTU is as follows:

- When the TCP source device receives an ICMP error message, it reduces the path MTU and starts an aging timer for the path MTU.
- After the aging timer expires, the source device uses a larger MSS in the MTU table, as described in RFC 1191.
- If no ICMP error message is received within two minutes, the source device increases the MSS again until the MSS negotiated during TCP three-way handshake is reached.

Prerequisites

Make sure all devices on a TCP connection are enabled to send ICMP error messages by using the **`ip unreachables enable`** command.

Procedure

1. Enter system view.

   `system-view`

2. Enable TCP path MTU discovery.

   `tcp path-mtu-discovery` [**`aging`** *`age-time`* | **`no-aging`**]

   By default, TCP path MTU discovery is disabled.

Enabling SYN Cookie

About this task

A TCP connection is established through a three-way handshake. An attacker can exploit this mechanism to mount SYN Flood attacks. The attacker sends a large number of SYN packets, but does not respond to the SYN ACK packets from the server. As a result, the server establishes a large number of TCP semi-connections and can no longer handle normal services.

SYN Cookie can protect the server from SYN Flood attacks. When the server receives a SYN packet, it responds with a SYN ACK packet without establishing a TCP semi-connection. The server establishes a TCP connection and enters ESTABLISHED state only when it receives an ACK packet from the client.

Procedure

1. Enter system view.

   `system-view`

2. Enable SYN Cookie.

   `tcp syn-cookie enable`

   By default, SYN Cookie is disabled.

Setting the TCP buffer size

1. Enter system view.

   ```
   system-view
   ```

2. Set the size of TCP receive/send buffer.

   ```
   tcp window window-size
   ```

   The default buffer size is 63 KB.

Setting TCP timers

About this task

You can set the following TCP timers:

- **SYN wait timer**—TCP starts the SYN wait timer after sending a SYN packet. Within the SYN wait timer if no response is received or the upper limit on TCP connection tries is reached, TCP fails to establish the connection.
- **FIN wait timer**—TCP starts the FIN wait timer when TCP changes the connection state to FIN_WAIT_2. If no FIN packet is received within the timer interval, TCP terminates the connection. If a FIN packet is received, TCP changes the connection state to TIME_WAIT. If a non-FIN packet is received, TCP restarts the timer, and tears down the connection when the timer expires.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Set the TCP SYN wait timer.

   ```
   tcp timer syn-timeout time-value
   ```

   By default, the TCP SYN wait timer is 75 seconds.

3. Set the TCP FIN wait timer.

   ```
   tcp timer fin-timeout time-value
   ```

   By default, the TCP FIN wait timer is 675 seconds.

Enabling carrying the TCP timestamp option in outgoing TCP packets

About this task

The TCP timestamp option in TCP packets is used to calculate the RTT between two communicating devices. In some networks, it is required to prevent the intermediate devices from obtaining the TCP timestamps in packets passing through. Then you can disable carrying the TCP timestamp option in outgoing packets on a device at either end.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable carrying the TCP timestamp option in outgoing TCP packets.

   ```
   tcp timestamps enable
   ```

   By default, the device adds the TCP timestamp option in outgoing TCP packets.

Display and maintenance commands for IP performance optimization

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display ICMP statistics.	**display icmp statistics** [**slot** *slot-number*]
Display IP packet statistics.	**display ip statistics** [**slot** *slot-number*]
Display brief information about RawIP connections.	**display rawip** [**slot** *slot-number*]
Display detailed information about RawIP connections.	**display rawip verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Display brief information about TCP connections.	**display tcp** [**slot** *slot-number*]
Display TCP traffic statistics.	**display tcp statistics** [**slot** *slot-number*]
Display detailed information about TCP connections.	**display tcp verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Display brief information about TCP proxy.	**display tcp-proxy slot** *slot-number*
Display the usage of non-well known ports for TCP proxy.	**display tcp-proxy port-info slot** *slot-number*
Display brief information about UDP connections.	**display udp** [**slot** *slot-number*]
Display UDP traffic statistics.	**display udp statistics** [**slot** *slot-number*]
Display detailed information about UDP connections.	**display udp verbose** [**slot** *slot-number* [**pcb** *pcb-index*]]
Clear IP packet statistics.	**reset ip statistics** [**slot** *slot-number*]
Clear TCP traffic statistics.	**reset tcp statistics**
Clear UDP traffic statistics.	**reset udp statistics**

Contents

Configuring multi-CPU packet distribution

About multi-CPU packet distribution

On a multi-CPU device, packets can be distributed among CPUs based on one of the following policies:

- **Flow-based policy**—Forwards packets of a flow to one CPU or multiple CPUs. The CPU processes flow packets by following the first-in first-out rule. The device supports using the source IP address, destination IP address, source port number, destination port number, and protocol number to identify a flow.
- **Packet-based policy**—Forwards packets in sequence to different CPUs, even though they are the same flow. This policy does not ensure packet order.

Restrictions and guidelines: Multi-CPU packet distribution configuration

If a service requires packets of a flow to be received by the same CPU, you must use the flow-based policy.

Specifying a packet distribution policy

1. Enter system view.

   `system-view`

2. Specify a packet distribution policy.

   `forwarding policy` { **`per-flow`** [**`three-tuple`** | **`mode`** { **`source-ip`** | **`destination-ip`** | **`source-port`** | **`destination-port`** }] | **`per-packet`** }

   By default, the device uses the flow-based policy that identifies a flow by source IP address, destination IP address, source port number, destination port number, and protocol number.

   Support for the **`three-tuple`** and **`mode`** keywords depends on the device model. For more information, see the command reference.

Contents

Displaying the adjacency table

About the adjacency table

The adjacency table stores information about directly connected neighbors for IP forwarding. The neighbor information in this chapter refers to non-Ethernet neighbor information.

This table is not user configurable. The neighbor information is generated, updated, and deleted by link layer protocols through negotiation (such as PPP dynamic negotiation) or through manual configuration. An adjacency entry includes the following information:

- Neighbor network layer address (next hop).
- Output interface.
- Link layer protocol type.
- Link layer address. This field is not available for PPP.

When forwarding an IP packet, the device performs the following tasks:

- Searches the FIB to find the output interface and next hop.
- Uses the output interface and next hop address to search the adjacency table for link layer forwarding information.

NOTE:

Ethernet and non-Ethernet neighbor information is stored and managed together.

Display and maintenance commands for adjacency table

Execute **display** commands in any view.

Task	Command
Display IPv4 adjacency table information.	**display adjacent-table** { **all** | **physical-interface** *interface-type interface-number* | **routing-interface** *interface-type interface-number* | **slot** *slot-number* } [**count** | **verbose**]
Display IPv6 adjacency table information.	**display ipv6 adjacent-table** { **all** | **physical-interface** *interface-type interface-number* | **routing-interface** *interface-type interface-number* | **slot** *slot-number* } [**count** | **verbose**]

Contents

Configuring Web caching

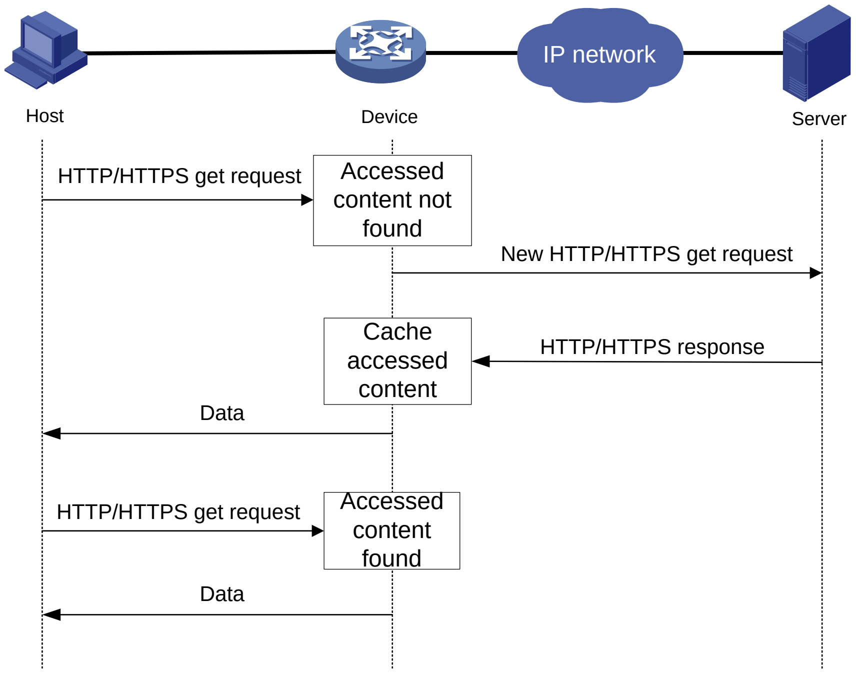
About Web caching

The Web caching feature saves specific webpage content requested by HTTP or HTTPS users from specific Web servers to files in the Web cache directory. If users access the same content before the content is aged out, the device retrieves the content from the files and sends the content to the users. Web caching can reduce the traffic between the device and the Web servers, lower Web content transmission costs, relieve pressure on Web servers, and improve user Web access speed.

Web caching mechanism

Figure 1 shows the Web caching mechanism.

Figure 1 Web caching mechanism



1. The host sends an HTTP or HTTPS GET request to the device.
2. The device identifies whether the Web caching feature is configured to cache Web content from the Web server.
   - If not, the device forwards the request directly to the Web server. The Web caching feature does not take part in the communication between the host and the Web server.
   - If yes, the device forwards the request to the Web caching process.
3. The Web caching process obtains the URL of the request and looks up its cache for the requested content.
   - If the requested content is found in the cache, the device retrieves the content and sends the content to the host. The HTTP or HTTPS GET operation is completed.
   - If the requested content is not found, the device re-constructs the request and sends the request to the Web server.
4. The Web server sends the requested content to the device.

5. The device identifies whether the Web caching feature is configured to cache the content.
    - If yes, the device uses the Web caching feature to cache the content, re-constructs a response, and sends the response to the host.
    - If not, the device re-constructs a response and sends the response to the host.

Web caching backup

The Web cache backup feature enhances the high availability of the Web caching service. You can specify a primary slot and a backup slot for Web caching. When the primary slot fails, the backup slot will take over to provide the Web caching service to ensure service continuity. When the primary slot recovers, the Web caching service switches back to the primary slot.

Basic concepts

- **Web cache files and Web cache directory**—The Web caching feature saves its operation data and the Web content to be cached to files in the Web cache directory. After the effective maximum total size for Web cache files is reached, the device deletes the oldest Web cache file to save the new Web cache file.
- **Web cache file aging time**—The Web cache file aging time is fixed at 30 days. When the device reboots or receives a request for the cached content in a file, it restarts the aging timer. If no users request the cached content before the aging timer expires, the device deletes the file.

Restrictions: Software version compatibility with Web caching

Web caching is supported only in RXX60P20 and later.

License requirement for Web caching

The Web caching feature requires a license. If you configure the feature without a license, the settings will be lost after a device reboot. For information about feature licensing, see *Fundamentals Configuration Guide*.

Restrictions and guidelines: Web caching configuration

Before configuring or modifying Web caching parameters, you must disable Web caching. After configuring or modifying Web caching parameters, enable Web caching again.

Web caching tasks at a glance

To configure Web caching, perform the following tasks:

1. Configuring Web caching parameters
2. Enabling Web caching

Configuring Web caching parameters

Restrictions and guidelines

Before configuring or modifying Web caching parameters, you must disable Web caching. After configuring or modifying Web caching parameters, enable Web caching again.

The Web cache directory for a Web cache view must reside on the same slot as the Web view.

Before specifying a Web cache directory, make sure all files in the upper-level directory are using a different name than the Web cache directory or have a file extension.

Make sure the storage medium where the Web cache directory resides has sufficient storage space. The directory typically needs a storage space of over 1 GB.

Prerequisites

To cache only Web content from certain Web servers, configure an IPv4 object group to identify the Web servers. For more information about the IPv4 object group feature, see object group configuration in *Security Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create and enter Web cache view.

   web-cache slot *slot-number*

3. Specify the types of the webpage files to be cached.

   cached-data { **apk** | **bmp** | **doc** | **docx** | **gif** | **gzip** | **ipa** | **jar** | **jpg** | **jpeg** | **mp4** | **pdf** | **png** | **ppt** | **pptx** | **rar** | **swf** | **tar** | **txt** | **xls** | **xlsx** | **zip** } *

   By default, no webpage file types are specified. The Web caching feature does not cache any types of files on webpages.

4. (Optional.) Specify webpage files to be cached.

   cached-file *file-name*

   By default, no webpage files to be cached are specified.

   If you specify both webpage files and types of the webpage files to be cached, the device caches matching webpage files of the specified types.

5. Set the Web caching listening port number.

   - Set the port number for Web caching to listen for HTTP packets.

     listen-port *port-number*

   - Set the port number for Web caching to listen for HTTPS packets.

     https listen-port *port-number*

   By default, Web caching uses port 80 to listen for HTTP packets and port 443 to listen for HTTPS packets.

   Make sure the specified TCP port number is not being used by any other services on the device. To display TCP port numbers in use, execute the **display tcp verbose** command.

6. (Optional.) Specify an IPv4 object group used to filter Web content to be cached.

   object-group [**source**] **ip** *object-group-name*

   By default, no IPv4 object group is specified for the Web caching feature to filter Web content. The Web caching feature caches Web content from all Web servers.

   To cache Web content requested by specific Web clients, specify the **source** keyword. To cache Web content sent from specific Web servers, do not specify the **source** keyword.

7. Set the primary Web cache directory.

   file-directory *directory*

   By default, the primary Web cache directory is not set.

8. (Optional.) Set the maximum total size of Web cache files.

   cache-limit *size*

   By default, the maximum total size of Web cache files is 4 GB.

   The maximum total size for Web cache files must be smaller than the maximum storage space size in the working directory.

9. (Optional.) Configure the Web caching backup feature.

   a. Specify the backup Web caching slot.

      backup slot *slot-number*

      By default, no backup Web caching slot is specified.

   b. Set the backup Web cache directory.

      file-directory backup *directory*

      By default, no backup Web cache directory is set.

Enabling Web caching

Restrictions and guidelines

Before enabling Web caching, you must configure the types of the webpage files to be cached and the Web cache directory.

Web caching will restart in one of the following situations:

- When HTTP-based Web caching is configured, enable HTTPS-based Web caching.
- When HTTPS-based Web caching is configured, enable HTTP-based Web caching.
- When both HTTP-based and HTTPS-based Web caching are configured, disable HTTP-based or HTTPS-based Web caching.

Procedure

1. Enter system view.

   system-view

2. Create and enter Web cache view.

   web-cache slot *slot-number*

3. Enable Web caching. Choose the options to configure as needed:

   - Enable HTTP-based Web caching.

     http enable

     By default, HTTP-based Web caching is disabled.

   - Enable HTTPS-based Web caching.

     https enable

     By default, HTTPS-based Web caching is disabled.

Display and maintenance commands for Web caching

Execute the **display** command in any view.

Task	Command
Display Web caching information.	**display web-cache** [**history** [**last** { **day** | **30-days** | **365-days** | **hour** | **minute** | **week** } | **verbose**]]

Troubleshooting Web caching

Web caching enabling failure

Symptom

Web caching failed to be enabled.

Analysis

The TCP port number specified for Web caching must belong to Web caching exclusively. If any other services on the device are using the same TCP port number, Web caching cannot be enabled.

Solution

To resolve the problem:

1. Execute the **display tcp verbose** command to display TCP port numbers being used.
2. Specify an unused TCP port number as the Web caching listening port number.

Web caching backup failure

Symptom

The device does not use the backup Web caching slot for Web caching after the primary Web caching slot reboots.

Analysis

If the storage medium of the backup Web caching slot is inaccessible or is not correctly configured, the Web caching service cannot be enabled on the backup Web caching slot.

Solution

To resolve the problem:

1. Execute the **dir** command to display information about the backup Web caching slot and then make sure the storage medium of the directory is accessible.

   For more information about the **dir** command, see file system management commands in *Fundamentals Command Reference.*
2. If the issue persists, contact NSFOCUS Support.

NSFOCUS Firewall Series

NF Layer 3—IP Routing

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for routing protocols, including basic IP routing, static routing, RIP, OSPF, IS-IS, BGP, policy-based routing, IPv6 static routing, IPv6 policy-based routing, RIPng, OSPFv3, Guard route, RIR and routing policies.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

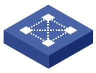
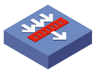
GUI conventions

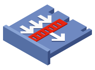
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring basic IP routing

This chapter focuses on unicast routing protocols. For more information about multicast routing protocols, see *IP Multicast Configuration Guide*.

About IP routing

IP routing directs IP packet forwarding on routers. Based on the destination IP address in the packet, a router looks up a route for the packet in a routing table and forwards the packet to the next hop. Routes are path information used to direct IP packets.

Routing table

A RIB contains the global routing information and related information, including route recursion, route redistribution, and route extension information. The router selects optimal routes from the routing table and puts them into the FIB table. It uses the FIB table to forward packets. For more information about the FIB table, see *Layer 3—IP Services Configuration Guide*.

Route categories

Table 1 categorizes routes by different criteria.

Table 1 Route categories

Criterion	Categories
Origin	• **Direct route**—A direct route is discovered by the data link protocol on an interface, and is also called an interface route. • **Static route**—A static route is manually configured by an administrator. • **Dynamic route**—A dynamic route is dynamically discovered by a routing protocol.
Destination	• **Network route**—The destination is a network. The subnet mask is less than 32 bits. • **Host route**—The destination is a host. The subnet mask is 32 bits.
Whether the destination is directly connected	• **Direct route**—The destination is directly connected. • **Indirect route**—The destination is indirectly connected.

Dynamic routing protocols

Static routes work well in small, stable networks. They are easy to configure and require fewer system resources. However, in networks where topology changes occur frequently, a typical practice is to configure a dynamic routing protocol. Compared with static routing, a dynamic routing protocol is complicated to configure, requires more router resources, and consumes more network resources.

Dynamic routing protocols dynamically collect and report reachability information to adapt to topology changes. They are suitable for large networks.

Dynamic routing protocols can be classified by different criteria, as shown in Table 2.

Table 2 Categories of dynamic routing protocols

Criterion	Categories
Operation scope	• **IGPs**—Work within an AS. Examples include RIP, OSPF, and IS-IS. • **EGPs**—Work between ASs. The most popular EGP is BGP.
Routing algorithm	• **Distance-vector protocols**—Examples include RIP and BGP. BGP is also considered a path-vector protocol. • **Link-state protocols**—Examples include OSPF and IS-IS.
Destination address type	• **Unicast routing protocols**—Examples include RIP, OSPF, BGP, and IS-IS. • **Multicast routing protocols**—Examples include PIM-SM and PIM-DM.
IP version	• **IPv4 routing protocols**—Examples include RIP, OSPF, BGP, and IS-IS. • **IPv6 routing protocols**—Examples include RIPng, OSPFv3, IPv6 BGP, and IPv6 IS-IS.

An AS refers to a group of routers that use the same routing policy and work under the same administration.

Route preference

Routing protocols, including static and direct routing, each by default have a preference. If they find multiple routes to the same destination, the router selects the route with the highest preference as the optimal route.

The preference of a direct route is always 0 and cannot be changed. You can configure a preference for each static route and each dynamic routing protocol. The following table lists the route types and default preferences. The smaller the value, the higher the preference.

Table 3 Route types and default route preferences

Route type	Preference
Direct route	0
Multicast static route	1
OSPF	10
IS-IS	15
Unicast static route	60
RIP	100
OSPF ASE	150
OSPF NSSA	150
Guard	254
IBGP	255
EBGP	255
Unknown (route from an untrusted source)	256

Load sharing

A routing protocol might find multiple optimal equal-cost routes to the same destination. You can use these routes to implement equal-cost multi-path (ECMP) load sharing.

Static routing, IPv6 static routing, RIP, RIPng, OSPF, OSPFv3, BGP, IPv6 BGP, IS-IS, and IPv6 IS-IS support ECMP load sharing.

Route backup

Route backup can improve network availability. Among multiple routes to the same destination, the route with the highest priority is the primary route and others are secondary routes.

The router forwards matching packets through the primary route. When the primary route fails, the route with the highest preference among the secondary routes is selected to forward packets. When the primary route recovers, the router uses it to forward packets.

Route recursion

To use a BGP, static, or RIP route that has an indirectly connected next hop, a router must perform route recursion to find the output interface to reach the next hop.

Link-state routing protocols, such as OSPF and IS-IS, do not need route recursion, because they obtain directly connected next hops through route calculation.

The RIB records and saves route recursion information, including brief information about related routes, recursive paths, and recursion depth.

Route redistribution

Route redistribution enables routing protocols to learn routing information from each other. A dynamic routing protocol can redistribute routes from other routing protocols, including direct and static routing. For more information, see the respective chapters on those routing protocols in this configuration guide.

The RIB records redistribution relationships of routing protocols.

Extension attribute redistribution

Extension attribute redistribution enables routing protocols to learn route extension attributes from each other, including BGP extended community attributes, OSPF area IDs, route types, and router IDs.

The RIB records extended attributes of each routing protocol and redistribution relationships of different routing protocol extended attributes.

Setting the maximum lifetime for routes in the RIB

About this task

Perform this task to prevent routes of a certain protocol from being aged out due to slow protocol convergence resulting from a large number of route entries or long GR period.

Restrictions and guidelines

The configuration takes effect at the next protocol or RIB process switchover.

Procedure (IPv4)

1. Enter system view.
   system-view
2. Enter RIB view.
   rib
3. Create the RIB IPv4 address family and enter its view.
   address-family ipv4
4. Set the maximum lifetime for IPv4 routes in the RIB.
   protocol *protocol* **lifetime** *seconds*

   By default, the maximum lifetime for routes in the RIB is 480 seconds.

Procedure (IPv6)

1. Enter system view.
   system-view
2. Enter RIB view.
   rib
3. Create the RIB IPv6 address family and enter its view.
   address-family ipv6
4. Set the maximum lifetime for IPv6 routes in the RIB.
   protocol *protocol* **lifetime** *seconds*

   By default, the maximum lifetime for routes in the RIB is 480 seconds.

Setting the maximum lifetime for routes in the FIB

About this task

When GR or NSR is disabled, FIB entries must be retained for some time after a protocol process switchover or RIB process switchover. When GR or NSR is enabled, FIB entries must be removed immediately after a protocol or RIB process switchover to avoid routing issues. Perform this task to meet such requirements.

Procedure (IPv4)

1. Enter system view.
   system-view
2. Enter RIB view.
   rib
3. Create the RIB IPv4 address family and enter its view.
   address-family ipv4
4. Set the maximum lifetime for IPv4 routes in the FIB.
   fib lifetime *seconds*

   By default, the maximum lifetime for routes in the FIB is 600 seconds.

Procedure (IPv6)

1. Enter system view.
   system-view
2. Enter RIB view.
   rib

3. Create the RIB IPv6 address family and enter its view.

   ```
   address-family ipv6
   ```

4. Set the maximum lifetime for IPv6 routes in the FIB.

   ```
   fib lifetime seconds
   ```

   By default, the maximum lifetime for routes in the FIB is 600 seconds.

Setting the maximum number of ECMP routes

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB680, NFNX3-HDB1080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB1780, NFNX3-HDB3080	No

Restrictions and guidelines

This configuration takes effect at reboot. Make sure the reboot does not impact your network.

Procedure (IPv4)

1. Enter system view.

   ```
   system-view
   ```

2. Set the maximum number of ECMP routes.

   ```
   max-ecmp-num number
   ```

   The default maximum number of ECMP routes varies by device model. For more information, see the command reference.

Procedure (IPv6)

1. Enter system view.

   ```
   system-view
   ```

2. Set the maximum number of ECMP routes.

   ```
   ipv6 max-ecmp-num number
   ```

   The default maximum number of ECMP routes varies by device model. For more information, see the command reference.

Configuring RIB NSR

About this task

When an active/standby switchover occurs, nonstop routing (NSR) backs up routing information from the active process to the standby process to avoid routing flapping and ensure forwarding continuity.

RIB NSR provides faster route convergence than protocol NSR during an active/standby switchover.

Restrictions and guidelines

Use this feature with protocol GR or NSR to avoid route timeouts and traffic interruption.

Procedure (IPv4)

1. Enter system view.

   system-view

2. Enter RIB view.

   rib

3. Create the RIB IPv4 address family and enter its view.

   address-family ipv4

4. Enable IPv4 RIB NSR.

   non-stop-routing

   By default, RIB NSR is disabled.

Procedure (IPv6)

1. Enter system view.

   system-view

2. Enter RIB view.

   rib

3. Create the RIB IPv6 address family and enter its view.

   address-family ipv6

4. Enable IPv6 RIB NSR.

   non-stop-routing

   By default, RIB NSR is disabled.

Configuring inter-protocol FRR

About this task

Inter-protocol fast reroute (FRR) enables fast rerouting between routes of different protocols. A backup next hop is automatically selected to reduce the service interruption time caused by unreachable next hops. When the next hop of the primary link fails, the traffic is redirected to the backup next hop.

Among the routes to the same destination in the RIB, a router adds the route with the highest preference to the FIB table. For example, if a static route and an OSPF route in the RIB have the same destination, the router adds the OSPF route to the FIB table by default. The next hop of the static route is selected as the backup next hop for the OSPF route. When the next hop of the OSPF route is unreachable, the backup next hop is used.

Restrictions and guidelines

This feature uses the next hop of a route from a different protocol as the backup next hop, which might cause loops.

Procedure (IPv4)

1. Enter system view.

   system-view

2. Enter RIB view.

   rib

3. Create the RIB IPv4 address family and enter its view.

   address-family ipv4

4. Enable IPv4 RIB inter-protocol FRR.

inter-protocol fast-reroute [**vpn-instance** *vpn-instance-name*]

By default, inter-protocol FRR is disabled.

If you do not specify a VPN instance, inter-protocol FRR is enabled for the public network.

Procedure (IPv6)

1. Enter system view.
   system-view
2. Enter RIB view.
   rib
3. Create the RIB IPv6 address family and enter its view.
   address-family ipv6
4. Enable IPv6 RIB inter-protocol FRR.
   inter-protocol fast-reroute [**vpn-instance** *vpn-instance-name*]
   By default, inter-protocol FRR is disabled.
   If you do not specify a VPN instance, inter-protocol FRR is enabled for the public network.

Display and maintenance commands for basic IP routing

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display routing table information.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] [**verbose**]
Display information about routes permitted by an IPv4 basic ACL.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] **acl** *ipv4-acl-number* [**verbose**]
Display information about routes to a specific destination address.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] *ip-address* [*mask-length* | *mask*] [**longer-match**] [**verbose**]
Display information about routes to a range of destination addresses.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] *ip-address1* **to** *ip-address2* [**verbose**]
Display information about routes permitted by an IP prefix list.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] **prefix-list** *prefix-list-name* [**verbose**]
Display information about routes installed by a protocol.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] **protocol** *protocol* [**inactive** | **verbose**]
Display IPv4 route statistics.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] **statistics**
Display brief IPv4 routing table information.	**display ip routing-table** [**vpn-instance** *vpn-instance-name*] **summary**
Display route attribute information in the IPv6 RIB.	**display ipv6 rib attribute** [*attribute-id*]

Task	Command
Display IPv6 RIB GR state information.	**display ipv6 rib graceful-restart**
Display next hop information in the IPv6 RIB.	**display ipv6 rib nib** [**self-originated**] [*nib-id*] [**verbose**] **display ipv6 rib nib protocol** *protocol* [**verbose**]
Display next hop information for IPv6 direct routes.	**display ipv6 route-direct nib** [*nib-id*] [**verbose**]
Display IPv6 routing table information.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] [**verbose**]
Display information about routes permitted by an IPv6 basic ACL.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] **acl** *ipv6-acl-number* [**verbose**]
Display information about routes to an IPv6 destination address.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] *ipv6-address* [*prefix-length*] [**longer-match**] [**verbose**]
Display information about routes to a range of IPv6 destination addresses.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] *ipv6-address1* **to** *ipv6-address2* [**verbose**]
Display information about routes permitted by an IPv6 prefix list.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] **prefix-list** *prefix-list-name* [**verbose**]
Display information about routes installed by an IPv6 protocol.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] **protocol** *protocol* [**inactive** | **verbose**]
Display IPv6 route statistics.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] **statistics**
Display brief IPv6 routing table information.	**display ipv6 routing-table** [**vpn-instance** *vpn-instance-name*] **summary**
Display route attribute information in the RIB.	**display rib attribute** [*attribute-id*]
Display RIB GR state information.	**display rib graceful-restart**
Display next hop information in the RIB.	**display rib nib** [**self-originated**] [*nib-id*] [**verbose**] **display rib nib protocol** *protocol* [**verbose**]
Display next hop information for direct routes.	**display route-direct nib** [*nib-id*] [**verbose**]
Clear IPv4 route statistics.	**reset ip routing-table statistics protocol** [**vpn-instance** *vpn-instance-name*] { *protocol* | **all** }
Clear IPv6 route statistics.	**reset ipv6 routing-table statistics protocol** [**vpn-instance** *vpn-instance-name*] { *protocol* | **all** }

Contents

Configuring static routing

About static routes

Static routes are manually configured. If a network's topology is simple, you only need to configure static routes for the network to work correctly.

Static routes cannot adapt to network topology changes. If a fault or a topological change occurs in the network, the network administrator must modify the static routes manually.

Configuring a static route

1. Enter system view.

   ```
   system-view
   ```

2. Configure a static route.

   Public network:

   ```
   ip route-static dest-address { mask-length | mask } { interface-type
   interface-number [ next-hop-address ] | next-hop-address |
   vpn-instance d-vpn-instance-name next-hop-address } [ permanent |
   track track-entry-number ] [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   By default, no static route is configured.

   You can associate Track with a static route to monitor the reachability of the next hops. For more information about Track, see *Network Management and Monitoring Configuration Guide*.

   VPN:

   ```
   ip route-static vpn-instance s-vpn-instance-name dest-address
   { mask-length | mask } { interface-type interface-number
   [ next-hop-address ] | next-hop-address [ public ] | vpn-instance
   d-vpn-instance-name next-hop-address } [ permanent | track
   track-entry-number ] [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   By default, no static route is configured.

   You can associate Track with a static route to monitor the reachability of the next hops. For more information about Track, see *Network Management and Monitoring Configuration Guide*.

3. (Optional.) Configure the default preference for static routes.

   ```
   ip route-static default-preference default-preference
   ```

   The default setting is 60.

Configuring a static route group

About this task

This task allows you to batch create static routes with different prefixes but the same output interface and next hop.

You can create a static route group, and specify the static group in the **ip route-static** command. All prefixes in the static route group will be assigned the next hop and output interface specified in the **ip route-static** command.

Procedure

1. Enter system view.

   system-view

2. Create a static route group and enter its view.

   ip route-static-group *group-name*

   By default, no static route group is configured.

3. Add a static route prefix to the static route group.

   prefix *dest-address* { *mask-length* | *mask* }

   By default, no static route prefix is added to the static route group.

4. Return to system view.

   quit

5. Configure a static route.

   Public network:

   ip route-static group *group-name* { *interface-type interface-number* [*next-hop-address*] | *next-hop-address* | **vpn-instance** *d-vpn-instance-name next-hop-address* } [**permanent** | **track** *track-entry-number*] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   VPN:

   ip route-static vpn-instance *s-vpn-instance-name* **group** *group-name* { *interface-type interface-number* [*next-hop-address*] | *next-hop-address* [**public**] | **vpn-instance** *d-vpn-instance-name next-hop-address* } [**permanent** | **track** *track-entry-number*] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   By default, no static route is configured.

Configuring the DHCP-designated default router as the next hop of a static route

About this task

After an interface obtains an IP address and gateway address through DHCP, the device automatically generates a static route with the interface as the output interface. The destination address of the static route is 0.0.0.0/0 and the next hop of the static route is the default router (the gateway address designated by the DHCP server). This static route cannot form ECMP routes with manually configured static routes. The device uses this static route to guide traffic forwarding only after the manually configured static routes become invalid.

Perform this task to use both the automatically generated static route and the manually configured static routes to guide traffic forwarding. The task is applicable when the device has dual egress WAN links.

This task enables the device to automatically generate a static route destined for the specified network with the DHCP-designated default router of the output interface as the next hop. This static route takes effect only after the output interface obtains an IP address and gateway address through DHCP, and becomes invalid upon the DHCP lease expiration. The next hop of this static route changes as the gateway address of the output interface changes. In addition, this static route can form ECMP routes with manually configured static routes.

Restrictions and guidelines

When you configure the next hop of a static route as the DHCP-designated default router, make sure the output interface of the static route is a broadcast interface.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure a static route and specify the default router designated by the DHCP server for the output interface as the next hop of the static route.

   Public network:

   ```
   ip route-static { dest-address { mask-length | mask } | group group-name }
   interface-type interface-number dhcp [ backup-interface
   interface-type interface-number [ backup-nexthop
   backup-nexthop-address ] [ permanent ] | permanent | track
   track-entry-number ] [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   VPN:

   ```
   ip route-static vpn-instance s-vpn-instance-name dest-address
   { mask-length | mask } interface-type interface-number dhcp
   [ backup-interface interface-type interface-number [ backup-nexthop
   backup-nexthop-address ] [ permanent ] | permanent | track
   track-entry-number ] [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   ```
   ip route-static vpn-instance s-vpn-instance-name group group-name
   interface-type interface-number dhcp [ backup-interface
   interface-type interface-number [ backup-nexthop
   backup-nexthop-address ] [ permanent ] ] [ preference preference ] [ tag
   tag-value ] [ description text ]
   ```

   By default, no static route is configured with the DHCP-designated default router as the next hop.

Deleting static routes

About this task

To delete a static route, use the **undo ip route-static** command. To delete all static routes including the default route, use the **delete static-routes all** command.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Delete all static routes.

   Public network:

   ```
   delete static-routes all
   ```

   VPN:

   ```
   delete vpn-instance vpn-instance-name static-routes all
   ```

CAUTION:

This command might interrupt network communication and cause packet forwarding failure. Before executing the command, make sure you fully understand the potential impact on the network.

Configuring BFD for static routes

IMPORTANT:

Enabling BFD for a flapping route could worsen the situation.

About BFD

BFD provides a general-purpose, standard, medium-, and protocol-independent fast failure detection mechanism. It can uniformly and quickly detect the failures of the bidirectional forwarding paths between two routers for protocols, such as routing protocols.

For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

Configuring BFD control packet mode

About this task

This mode uses BFD control packets to detect the status of a link bidirectionally at a millisecond level.

BFD control packet mode can be applied to static routes with a direct next hop or with an indirect next hop.

Restrictions and guidelines for BFD control packet mode

If you use BFD control packet mode at the local end, you must use this mode also at the peer end.

Configuring BFD control packet mode for a static route (direct next hop)

1. Enter system view.

   `system-view`

2. Configure BFD control packet mode for a static route.

   Public network:

   ip route-static *dest-address* { *mask-length* | *mask* } *interface-type interface-number* { **dhcp** | *next-hop-address* } **bfd** { **control-packet** | **static** *session-name* } [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   VPN:

   ip route-static vpn-instance *s-vpn-instance-name dest-address* { *mask-length* | *mask* } *interface-type interface-number* { **dhcp** | *next-hop-address* } **bfd** { **control-packet** | **static** *session-name* } [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   By default, BFD control packet mode for a static route is not configured.

Configuring BFD control packet mode for a static route (indirect next hop)

1. Enter system view.

   `system-view`

2. Configure BFD control packet mode for a static route.

   Public network:

   ip route-static *dest-address* { *mask-length* | *mask* } { *next-hop-address* **bfd control-packet bfd-source** *ip-address* | **vpn-instance**

```
d-vpn-instance-name next-hop-address bfd { control-packet bfd-source
ip-address | static session-name } } [ preference preference ] [ tag
tag-value ] [ description text ]
```

VPN:

```
ip route-static vpn-instance s-vpn-instance-name dest-address
{ mask-length | mask } { next-hop-address bfd { control-packet
bfd-source ip-address | static session-name } | vpn-instance
d-vpn-instance-name next-hop-address bfd { control-packet bfd-source
ip-address | static session-name } } [ preference preference ] [ tag
tag-value ] [ description text ]
```

By default, BFD control packet mode for a static route is not configured.

Configuring BFD echo packet mode

About this task

With BFD echo packet mode enabled for a static route, the output interface sends BFD echo packets to the destination device, which loops the packets back to test the link reachability.

Restrictions and guidelines

You do not need to configure BFD echo packet mode at the peer end.

Do not use BFD for a static route with the output interface in spoofing state.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the source address of echo packets.

   ```
   bfd echo-source-ip ip-address
   ```

   By default, the source address of echo packets is not configured.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Configure BFD echo packet mode for a static route.

   Public network:

   ```
   ip route-static dest-address { mask-length | mask } interface-type
   interface-number { dhcp | next-hop-address } bfd { echo-packet | static
   session-name } [ preference preference ] [ tag tag-value ] [ description
   text ]
   ```

   VPN:

   ```
   ip route-static vpn-instance s-vpn-instance-name dest-address
   { mask-length | mask } interface-type interface-number { dhcp |
   next-hop-address } bfd { echo-packet | static session-name }
   [ preference preference ] [ tag tag-value ] [ description text ]
   ```

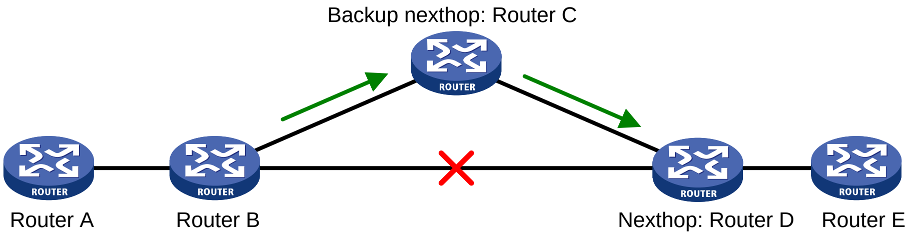
   By default, BFD echo packet mode for a static route is not configured.

Configuring static route FRR

About static route FRR

A link or router failure on a path can cause packet loss. Static route fast reroute (FRR) enables fast rerouting to minimize the impact of link or node failures.

Figure 1 Network diagram



As shown in Figure 1, upon a link failure, packets are directed to the backup next hop to avoid traffic interruption. You can either specify a backup next hop for FRR or enable FRR to automatically select a backup next hop (which must be configured in advance).

Restrictions and guidelines for static route FRR

Do not use static route FRR and BFD (for a static route) at the same time.

Equal-cost routes do not support static route FRR.

Besides the configured static route for FRR, the device must have another route to reach the destination. When the state of the primary link (with Layer 3 interfaces staying up) changes from bidirectional to unidirectional or down, static route FRR quickly redirects traffic to the backup next hop. When the Layer 3 interfaces of the primary link are down, static route FRR temporarily redirects traffic to the backup next hop. In addition, the device searches for another route to reach the destination and redirects traffic to the new path if a route is found. If no route is found, traffic interruption occurs.

Configuring static route FRR by specifying a backup next hop

Restrictions and guidelines

A static route does not take effect when the backup output interface is unavailable.

To change the backup output interface or next hop, you must first remove the current setting. The backup output interface and next hop must be different from the primary output interface and next hop.

Procedure

1. Enter system view.

   system-view

2. Configure static route FRR.

   Public network:

   ip route-static *dest-address* { *mask-length* | *mask* } *interface-type interface-number* [{ **dhcp** | *next-hop-address* } [**backup-interface** *interface-type interface-number* [**backup-nexthop** *backup-nexthop-address*]]] [**permanent**] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   VPN:

   ip route-static vpn-instance *s-vpn-instance-name dest-address* { *mask-length* | *mask* } *interface-type interface-number* [{ **dhcp** | *next-hop-address* } [**backup-interface** *interface-type interface-number* [**backup-nexthop** *backup-nexthop-address*]]] [**permanent**] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   By default, static route FRR is disabled.

Configuring static route FRR to automatically select a backup next hop

1. Enter system view.

   system-view

2. Configure static route FRR to automatically select a backup next hop.

   ip route-static fast-reroute auto

   By default, static route FRR is disabled from automatically selecting a backup next hop.

Enabling BFD echo packet mode for static route FRR

About this task

By default, static route FRR uses ARP to detect primary link failures. Perform this task to enable static route FRR to use BFD echo packet mode for fast failure detection on the primary link.

Procedure

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interface's IP address.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Enable BFD echo packet mode for static route FRR.

   ip route-static primary-path-detect bfd echo

   By default, BFD echo packet mode for static route FRR is disabled.

Display and maintenance commands for static routing

Execute **display** commands in any view.

Task	Command
Display static route information.	**display ip routing-table protocol static** [**inactive** | **verbose**]
Display static route next hop information.	**display route-static nib** [*nib-id*] [**verbose**]
Display static routing table information.	**display route-static routing-table** [**vpn-instance** *vpn-instance-name*] [*ip-address* { *mask-length* | *mask* }]

For more information about the **display ip routing-table protocol static** [**inactive** | **verbose**] command, see basic IP routing in *Layer 3—IP Routing Command Reference*.

Configuring a default route

A default route is used to forward packets that do not match any specific routing entry in the routing table. Without a default route, packets that do not match any routing entries are discarded and an ICMP destination-unreachable packet is sent to the source.

A default route can be configured in either of the following ways:

- The network administrator can configure a default route with both destination and mask being 0.0.0.0. For more information, see "Configuring static routing."
- Some dynamic routing protocols (such as OSPF and RIP) can generate a default route. For example, an upstream router running OSPF can generate a default route and advertise it to other routers. These routers install the default route with the next hop being the upstream router. For more information, see the respective chapters on these routing protocols in this configuration guide.

Contents

Configuring IPv6 static routing

About IPv6 static routing

Static routes are manually configured and cannot adapt to network topology changes. If a fault or a topological change occurs in the network, the network administrator must modify the static routes manually. IPv6 static routing works well in a simple IPv6 network.

Configuring an IPv6 static route

1. Enter system view.

   ```
   system-view
   ```

2. Configure an IPv6 static route.

   Public network:

   ```
   ipv6 route-static ipv6-address prefix-length { interface-type
   interface-number [ next-hop-address ] | next-hop-address |
   vpn-instance d-vpn-instance-name next-hop-address } [ permanent |
   track track-entry-number ] [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   By default, no IPv6 static route is configured.

   VPN:

   ```
   ipv6 route-static vpn-instance s-vpn-instance-name ipv6-address
   prefix-length { interface-type interface-number [ next-hop-address ] |
   next-hop-address [ public ] | vpn-instance d-vpn-instance-name
   next-hop-address } [ permanent | track track-entry-number ]
   [ preference preference ] [ tag tag-value ] [ description text ]
   ```

   ```
   ipv6 route-static vpn-instance s-vpn-instance-name ipv6-address
   prefix-length nexthop-address [ public ] [ permanent ] [ preference
   preference ] [ tag tag-value ] [ description text ]
   ```

   By default, no IPv6 static route is configured.

3. (Optional.) Set the default preference for IPv6 static routes.

   ```
   ipv6 route-static default-preference default-preference
   ```

   The default setting is 60.

Deleting IPv6 static routes

About this task

To delete an IPv6 static route, use the **undo ipv6 route-static** command. To delete all IPv6 static routes including the default route, use the **delete ipv6 static-routes all** command.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Delete all IPv6 static routes, including the default route.

   ```
   delete ipv6 [ vpn-instance vpn-instance-name ] static-routes all
   ```

△ **CAUTION:**

This command might interrupt network communication and cause packet forwarding failure. Before executing the command, make sure you fully understand the potential impact on the network.

Configuring BFD for IPv6 static routes

About BFD for IPv6 static routes

BFD provides a general purpose, standard, and medium- and protocol-independent fast failure detection mechanism. It can uniformly and quickly detect the failures of the bidirectional forwarding paths between two routers for protocols, such as routing protocols. BFD for IPv6 static routes tests the reachability of the next hop for each IPv6 static route. If a next hop is unreachable, BFD deletes the associated IPv6 static route.

For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines for BFD

When you configure BFD for IPv6 static routes, follow these restrictions and guidelines:

- If you specify a source IPv6 address for BFD packets on the local device, you must specify that IPv6 address as the next hop IPv6 address on the peer device.
- If you specify a non-P2P output interface and a direct next hop, specify the **bfd-source** *ipv6-address* option as a best practice. Make sure the source IPv6 address of BFD packets meets the following requirements:
    - The address is the same as the IPv6 address of the output interface.
    - The address is on the same network segment as the next hop IPv6 address of the same type.

      For example, if the next hop IPv6 address is a link-local address, the source IPv6 address of BFD packets must also be a link-local address.
- Enabling BFD for a flapping route could worsen the situation.

Configuring BFD control packet mode

About this task

This mode uses BFD control packets to detect the status of a link bidirectionally at a millisecond level.

BFD control packet mode can be applied to IPv6 static routes with a direct next hop or with an indirect next hop.

Restrictions and guidelines for BFD control packet mode

If you configure BFD control packet mode at the local end, you must also configure this mode at the peer end.

Configuring BFD control packet mode for an IPv6 static route (direct next hop)

1. Enter system view.

   system-view
2. Configure BFD control packet mode for an IPv6 static route.

ipv6 route-static [**vpn-instance** *s-vpn-instance-name*] *ipv6-address prefix-length interface-type interface-number next-hop-address* **bfd control-packet** [**bfd-source** *ipv6-address*] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

By default, BFD control packet mode for an IPv6 static route is not configured.

Configuring BFD control packet mode for an IPv6 static route (indirect next hop)

1. Enter system view.

   system-view
2. Configure BFD control packet mode for an IPv6 static route.

   ipv6 route-static [**vpn-instance** *s-vpn-instance-name*] *ipv6-address prefix-length* [**vpn-instance** *d-vpn-instance-name*] { *next-hop-address* **bfd control-packet bfd-source** *ipv6-address* } [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   By default, BFD control packet mode for an IPv6 static route is not configured.

Configuring BFD echo packet mode

About this task

With BFD echo packet mode enabled for a static route, the output interface sends BFD echo packets to the destination device, which loops the packets back to test the link reachability.

Restrictions and guidelines

You do not need to configure BFD echo packet mode at the peer end.

Do not use BFD for a static route with the output interface in spoofing state.

Procedure

1. Enter system view.

   system-view
2. Configure the source address of echo packets.

   bfd echo-source-ipv6 *ipv6-address*

   By default, the source address of echo packets is not configured.

   The source address of echo packets must be a global unicast address.

   For more information about this command, see *Network Management and Monitoring Command Reference*.
3. Configure BFD echo packet mode for an IPv6 static route.

   ipv6 route-static [**vpn-instance** *s-vpn-instance-name*] *ipv6-address prefix-length interface-type interface-number next-hop-address* **bfd echo-packet** [**bfd-source** *ipv6-address*] [**preference** *preference*] [**tag** *tag-value*] [**description** *text*]

   By default, BFD echo packet mode for an IPv6 static route is not configured.

   The next hop IPv6 address must be a global unicast address.

Display and maintenance commands for IPv6 static routing

Execute **display** commands in any view.

Task	Command
Display IPv6 static route next hop information.	**display ipv6 route-static nib** [*nib-id*] [**verbose**]
Display IPv6 static routing table information.	**display ipv6 route-static routing-table** [**vpn-instance** *vpn-instance-name*] [*ipv6-address prefix-length*]
Display IPv6 static route information.	**display ipv6 routing-table protocol static** [**inactive** | **verbose**]

For more information about the **display ipv6 routing-table protocol static** [**inactive** | **verbose**] command, see basic IP routing in *Layer 3—IP Routing Command Reference*.

Configuring an IPv6 default route

A default IPv6 route is used to forward packets that match no entry in the routing table.

A default IPv6 route can be configured in either of the following ways:

- The network administrator can configure a default route with a destination prefix of **::/0**. For more information, see "Configuring IPv6 static routing."
- Some dynamic routing protocols (such as OSPFv3, IPv6 IS-IS, and RIPng) can generate a default IPv6 route. For example, an upstream router running OSPFv3 can generate a default IPv6 route and advertise it to other routers. These routers install the default IPv6 route with the next hop being the upstream router. For more information, see the respective chapters on those routing protocols in this configuration guide.

Contents

Configuring RIP

About RIP

Routing Information Protocol (RIP) is a distance-vector IGP suited to small-sized networks. It employs UDP to exchange route information through port 520.

RIP routing metrics

RIP uses a hop count to measure the distance to a destination. The hop count from a router to a directly connected network is 0. The hop count from a router to a directly connected router is 1. To limit convergence time, RIP restricts the value range of the metric from 0 to 15. A destination with a metric value of 16 (or greater) is considered unreachable. For this reason, RIP is not suitable for large-sized networks.

RIP route entries

RIP stores routing entries in a database. Each routing entry contains the following elements:

- **Destination address**—IP address of a destination host or a network.
- **Next hop**—IP address of the next hop.
- **Egress interface**—Egress interface of the route.
- **Metric**—Cost from the local router to the destination.
- **Route time**—Time elapsed since the last update. The time is reset to 0 when the routing entry is updated.
- **Route tag**—Used for route control. For more information, see "Configuring routing policies."

RIP operation

RIP works as follows:

1. RIP sends request messages to neighboring routers. Neighboring routers return response messages that contain their routing tables.
2. RIP uses the received responses to update the local routing table and sends triggered update messages to its neighbors. All RIP routers on the network do this to learn latest routing information.
3. RIP periodically sends the local routing table to its neighbors. After a RIP neighbor receives the message, it updates its routing table, selects optimal routes, and sends an update to other neighbors. RIP ages routes to keep only valid routes.

Routing loop prevention

RIP uses the following mechanisms to prevent routing loops:

- **Counting to infinity**—A destination with a metric value of 16 is considered unreachable. When a routing loop occurs, the metric value of a route will increment to 16 to avoid endless looping.
- **Triggered updates**—RIP immediately advertises triggered updates for topology changes to reduce the possibility of routing loops and to speed up convergence.
- **Split horizon**—Disables RIP from sending routes through the interface where the routes were learned to prevent routing loops and save bandwidth.

- **Poison reverse**—Enables RIP to set the metric of routes received from a neighbor to 16 and sends these routes back to the neighbor. The neighbor can delete such information from its routing table to prevent routing loops.

RIP versions

There are two RIP versions, RIPv1 and RIPv2.

RIPv1 is a classful routing protocol. It advertises messages only through broadcast. RIPv1 messages do not carry mask information, so RIPv1 can only recognize natural networks such as Class A, B, and C. For this reason, RIPv1 does not support discontiguous subnets.

RIPv2 is a classless routing protocol. It has the following advantages over RIPv1:

- Supports route tags to implement flexible route control through routing policies.
- Supports masks, route summarization, and CIDR.
- Supports designated next hops to select the best ones on broadcast networks.
- Supports multicasting route updates so only RIPv2 routers can receive these updates to reduce resource consumption.
- Supports plain text authentication and MD5 authentication to enhance security.

RIPv2 supports two transmission modes: broadcast and multicast. Multicast is the default mode using 224.0.0.9 as the multicast address. An interface operating in RIPv2 broadcast mode can also receive RIPv1 messages.

Protocols and standards

- RFC 1058, *Routing Information Protocol*
- RFC 1723, *RIP Version 2 - Carrying Additional Information*
- RFC 1721, *RIP Version 2 Protocol Analysis*
- RFC 1722, *RIP Version 2 Protocol Applicability Statement*
- RFC 1724, *RIP Version 2 MIB Extension*
- RFC 2082, *RIPv2 MD5 Authentication*
- RFC 2091, *Triggered Extensions to RIP to Support Demand Circuits*
- RFC 2453, *RIP Version 2*

RIP tasks at a glance

To configure RIP, perform the following tasks:

1. Configuring basic RIP
   a. Enabling RIP
   b. (Optional.) Controlling RIP reception and advertisement on interfaces
   c. (Optional.) Configuring a RIP version
   d. Specifying a RIP neighbor

      To enable RIP on a link that does not support broadcast or multicast, you must manually specify a RIP neighbor.
2. (Optional.) Configuring RIP route control
   - Configuring an additional routing metric
   - Configuring RIPv2 route summarization
   - Disabling host route reception

   - Advertising a default route
   - Configuring received/redistributed route filtering
   - Setting a preference for RIP
   - Configuring RIP route redistribution
3. (Optional.) Tuning and optimizing RIP networks
   - Setting RIP timers
   - Enabling split horizon and poison reverse
   - Setting the maximum number of RIP ECMP routes
   - Setting the RIP triggered update interval
   - Configuring the RIP packet sending rate
   - Setting the maximum length of RIP packets
4. (Optional.) Configuring RIP network management
5. Enhancing RIP availability
   - Configuring RIP GR
   - Enabling RIP NSR
   - Configuring BFD for RIP
   - Configuring RIP FRR
6. (Optional.) Enhancing RIP security
   - Enabling zero field check for incoming RIPv1 messages
   - Enabling source IP address check for incoming RIP updates
   - Configuring RIPv2 message authentication

Configuring basic RIP

Restrictions and guidelines for configuring basic RIP

To enable multiple RIP processes on a router, you must specify an ID for each process. A RIP process ID has only local significance. Two RIP routers having different process IDs can also exchange RIP packets.

Enabling RIP

About this task

You can enable RIP on a network and specify a wildcard mask for the network. After that, only the interface attached to the network runs RIP.

Restrictions and guidelines

If you configure RIP settings in interface view before enabling RIP, the settings do not take effect until RIP is enabled.

If a physical interface is attached to multiple networks, you cannot advertise these networks in different RIP processes.

You cannot enable multiple RIP processes on a physical interface.

The **rip enable** command takes precedence over the **network** command.

Enabling RIP on a network

1. Enter system view.

system-view

2. Enable RIP and enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

   By default, RIP is disabled.

3. Enable RIP on a network.

   network *network-address* [*wildcard-mask*]

   By default, RIP is disabled on a network.

   The **network** 0.0.0.0 command can enable RIP on all interfaces in a single process, but does not apply to multiple RIP processes.

Enabling RIP on an interface

1. Enter system view.

   system-view

2. Enable RIP and enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

   By default, RIP is disabled.

3. Return to system view.

   quit

4. Enter interface view.

   interface *interface-type interface-number*

5. Enable RIP on the interface.

   rip *process-id* **enable** [**exclude-subip**]

   By default, RIP is disabled on an interface.

Controlling RIP reception and advertisement on interfaces

About this task

You can perform this task to configure the following features:

- Suppressing an interface. The suppressed interface can receive RIP messages but cannot send RIP messages.
- Disabling an interface from sending RIP messages.
- Disabling an interface from receiving RIP messages.

Restrictions and guidelines for RIP reception and advertisement control on interfaces

An interface suppressed by using the **silent-interface** command can only receive RIP messages. It cannot send RIP messages. You can use the **silent-interface all** command to suppress all interfaces. The **silent-interface** command takes precedence over the **rip input** and **rip output** commands.

Suppressing an interface

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Suppress an interface.

   silent-interface { *interface-type interface-number* | **all** }

   By default, all RIP-enabled interfaces can send RIP messages.

The suppressed interface can still receive RIP messages and respond to unicast requests containing unknown ports.

Disabling an interface from receiving RIP messages

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Disable an interface from receiving RIP messages.

   `undo rip input`

   By default, a RIP-enabled interface can receive RIP messages.

Disabling an interface from sending RIP messages

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Disable an interface from sending RIP messages.

   `undo rip output`

   By default, a RIP-enabled interface can send RIP messages.

Configuring a RIP version

About this task

You can configure a global RIP version in RIP view or an interface-specific RIP version in interface view.

An interface preferentially uses the interface-specific RIP version. If no interface-specific version is specified, the interface uses the global RIP version. If neither a global nor interface-specific RIP version is configured, the interface sends RIPv1 broadcasts and can receive the following:

- RIPv1 broadcasts and unicasts.
- RIPv2 broadcasts, multicasts, and unicasts.

Procedure

1. Enter system view.

   `system-view`

2. Specify a RIP version.
   - Execute the following commands in sequence to specify a global RIP version:

     `rip` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

     `version` { **`1`** | **`2`** }

     By default, no global version is specified. An interface sends RIPv1 broadcasts, and can receive RIPv1 broadcasts and unicasts, and RIPv2 broadcasts, multicasts, and unicasts.
   - Execute the following commands in sequence to specify a RIP version on an interface:

     `interface` *`interface-type interface-number`*

     `rip version` { **`1`** | **`2`** [**`broadcast`** | **`multicast`**] }

     By default, no interface-specific RIP version is specified. The interface sends RIPv1 broadcasts, and can receive RIPv1 broadcasts and unicasts, and RIPv2 broadcasts, multicasts, and unicasts.

Specifying a RIP neighbor

About this task

Typically RIP messages are sent in broadcast or multicast. To enable RIP on a link that does not support broadcast or multicast, you must manually specify a RIP neighbor.

Restrictions and guidelines

As a best practice, do not use the **peer** *ip-address* command to specify a directly connected neighbor. The neighbor might receive a route update in both unicast and multicast (or broadcast) messages from the device.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIP view.

   `rip` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

3. Specify a RIP neighbor.

   `peer` *`ip-address`*

   By default, RIP does not unicast updates to any peer.

4. Disable source IP address check on inbound RIP updates.

   `undo validate-source-address`

   By default, source IP address check is enabled on inbound RIP updates.

   If the specified neighbor is not directly connected, disable source address check on incoming updates.

Configuring RIP route control

Configuring an additional routing metric

About this task

An additional routing metric (hop count) can be added to the metric of an inbound or outbound RIP route.

- An outbound additional metric is added to the metric of a sent route, and it does not change the route's metric in the routing table.
- An inbound additional metric is added to the metric of a received route before the route is added into the routing table, and the route's metric is changed. If the sum of the additional metric and the original metric is greater than 16, the metric of the route is 16.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify an inbound additional routing metric.

   `rip metricin` [**`route-policy`** *`route-policy-name`*] *`value`*

   By default, the additional metric of an inbound route is 0.

4. Specify an outbound additional routing metric.

rip metricout [**route-policy** *route-policy-name*] *value*

By default, the additional metric of an outbound route is 1.

Configuring RIPv2 route summarization

About this task

Perform this task to summarize contiguous subnets into a summary network and sends the network to neighbors. The smallest metric among all summarized routes is used as the metric of the summary route.

You can use the following methods to summarize routes in RIPv2:

- **Automatic summarization**—Configure RIPv2 to generate a natural network for contiguous subnets. For example, suppose there are three subnet routes 10.1.1.0/24, 10.1.2.0/24, and 10.1.3.0/24. Automatic summarization automatically creates and advertises a summary route 10.0.0.0/8 instead of the more specific routes.
- **Manual summarization**—Manually configure a summary route. RIPv2 advertises the summary route rather than more specific routes. For example, suppose contiguous subnets routes 10.1.1.0/24, 10.1.2.0/24, and 10.1.3.0/24 exist in the routing table. You can create a summary route 10.1.0.0/16 on GigabitEthernet 1/0/1 to advertise the summary route instead of the more specific routes. By default, natural masks are used to advertise summary routes. To manually configure a summary route on an interface, you must first disable RIPv2 automatic route summarization.

Restrictions and guidelines

To prevent loops caused by route summarization, create a black hole route by specifying interface NULL 0 as the output interface of the summary route. Packets that match the black hole route are dropped.

Enabling RIPv2 automatic route summarization

1. Enter system view.

   system-view
2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]
3. Enable RIPv2 automatic route summarization.

   summary

   By default, RIPv2 automatic route summarization is enabled.

   If subnets in the routing table are not contiguous, disable automatic route summarization to advertise more specific routes.

Advertising a summary route

1. Enter system view.

   system-view
2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]
3. Disable RIPv2 automatic route summarization.

   undo summary

   By default, RIPv2 automatic route summarization is enabled.
4. Return to system view.

   quit
5. Enter interface view.

interface *interface-type interface-number*

6. Configure a summary route.

   rip summary-address *ip-address* { *mask-length* | *mask* }

   By default, no summary route is configured.

Disabling host route reception

About this task

This task disables RIPv2 from receiving host routes from the same network to save network resources. This feature does not apply to RIPv1.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Disable RIP from receiving host routes.

   undo host-route

   By default, RIP receives host routes.

Advertising a default route

About this task

You can advertise a default route on all RIP interfaces in RIP view or on a specific RIP interface in interface view. The interface view setting takes precedence over the RIP view settings.

To disable an interface from advertising a default route, use the **rip default-route no-originate** command on the interface.

The router enabled to advertise a default route does not accept default routes from RIP neighbors.

Procedure

1. Enter system view.

   system-view

2. Advertise a default route.

   - Execute the following commands in sequence to configure RIP to advertise a default route:

     rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

     default-route { **only** | **originate** } [**cost** *cost-value* | **route-policy** *route-policy-name*] *

     By default, RIP does not advertise a default route.

   - Execute the following commands in sequence to configure a RIP interface to advertise a default route:

     interface *interface-type interface-number*

     rip default-route { { **only** | **originate** } [**cost** *cost-value* | **route-policy** *route-policy-name*] * | **no-originate** }

     By default, a RIP interface can advertise a default route if the RIP process is enabled to advertise a default route.

Configuring received/redistributed route filtering

About this task

This task allows you to create a policy to filter received or redistributed routes that match specific criteria such as an ACL or IP prefix list.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIP view.

   ```
   rip [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Configure the filtering of received routes.

   ```
   filter-policy { ipv4-acl-number | gateway prefix-list-name |
   prefix-list prefix-list-name [ gateway prefix-list-name ] } import
   [ interface-type interface-number ]
   ```

   By default, the filtering of received routes is not configured.

   This command filters received routes. Filtered routes are not installed into the routing table or advertised to neighbors.

4. Configure the filtering of redistributed routes.

   ```
   filter-policy { ipv4-acl-number | prefix-list prefix-list-name }
   export [ interface-type interface-number | bgp | direct | [ isis | ospf |
   rip ] [ process-id ] | static ]
   ```

   By default, the filtering of redistributed routes is not configured.

   This command filters redistributed routes, including routes redistributed with the **import-route** command.

Setting a preference for RIP

About this task

If multiple IGPs find routes to the same destination, the route found by the IGP that has the highest priority is selected as the optimal route. Perform this task to assign a preference to RIP. The smaller the preference value, the higher the priority.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIP view.

   ```
   rip [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Set a preference for RIP.

   ```
   preference { preference | route-policy route-policy-name } *
   ```

   The default preference for RIP is 100.

Configuring RIP route redistribution

About this task

Perform this task to configure RIP to redistribute routes from other routing protocols, including OSPF, IS-IS, BGP, static, and direct.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Configure route redistribution.

   ○ Redistribute routes from BGP.

   import-route bgp [*as-number*] [**allow-ibgp**] [**cost** *cost-value* | **route-policy** *route-policy-name* | **tag** *tag*] *

   ○ Redistribute direct or static routes.

   import-route { **direct** | **static** } [**cost** *cost-value* | **route-policy** *route-policy-name* | **tag** *tag*] *

   ○ Redistribute routes from IS-IS, OSPF, or other RIP processes.

   import-route { **isis** | **ospf** | **rip** } [*process-id* | **all-processes**] [**allow-direct** | **cost** *cost-value* | **route-policy** *route-policy-name* | **tag** *tag*] *

   By default, RIP does not redistribute routes.

   This command can redistribute only active routes. To view active routes, use the **display ip routing-table protocol** command.

4. (Optional.) Set a default cost for redistributed routes.

   default cost *cost-value*

   The default cost for redistributed routes is 0.

Tuning and optimizing RIP networks

Setting RIP timers

About this task

You can change the RIP network convergence speed by adjusting the following RIP timers:

- **Update timer**—Specifies the interval between route updates.
- **Timeout timer**—Specifies the route aging time. If no update for a route is received within the aging time, the metric of the route is set to 16.
- **Suppress timer**—Specifies how long a RIP route stays in suppressed state. When the metric of a route is 16, the route enters the suppressed state. A suppressed route can be replaced by an updated route that is received from the same neighbor before the suppress timer expires and has a metric less than 16.
- **Garbage-collect timer**—Specifies the interval from when the metric of a route becomes 16 to when it is deleted from the routing table. RIP advertises the route with a metric of 16. If no update is announced for that route before the garbage-collect timer expires, the route is deleted from the routing table.

Restrictions and guidelines

To avoid unnecessary traffic or route flapping, configure identical RIP timer settings on RIP routers.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set RIP timers.

   timers { **garbage-collect** *garbage-collect-value* | **suppress** *suppress-value* | **timeout** *timeout-value* | **update** *update-value* } *

   The default settings are as follows:

   - The garbage-collect timer is 120 seconds.
   - The suppress timer is 120 seconds.
   - The timeout timer is 180 seconds.
   - The update timer is 30 seconds.

Enabling split horizon and poison reverse

About this task

The split horizon and poison reverse features can prevent routing loops.

- Split horizon disables RIP from sending routes through the interface where the routes were learned to prevent routing loops between adjacent routers.
- Poison reverse allows RIP to send routes through the interface where the routes were learned. The metric of these routes is always set to 16 (unreachable) to avoid routing loops between neighbors.

Restrictions and guidelines

If both split horizon and poison reverse are configured, only the poison reverse feature takes effect.

Enabling split horizon

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable split horizon.

   rip split-horizon

   By default, split horizon is enabled.

Enabling poison reverse

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable poison reverse.

   rip poison-reverse

   By default, poison reverse is disabled.

Setting the maximum number of RIP ECMP routes

About this task

Perform this task to use ECMP routes to share the load of the RIP network.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the maximum number of RIP ECMP routes.

   maximum load-balancing *number*

   By default, the maximum number of RIP ECMP routes equals the maximum number of ECMP routes supported by the system.

Setting the RIP triggered update interval

About this task

Perform this task to avoid network overhead and reduce system resource consumption caused by frequent RIP triggered updates.

You can use the **timer triggered** command to set the maximum interval, minimum interval, and incremental interval for sending RIP triggered updates.

- For a stable network, the *minimum-interval* is used.
- If network changes become frequent, the incremental interval *incremental-interval* is used to extend the triggered update sending interval until the *maximum-interval* is reached.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the RIP triggered update interval.

   timer triggered *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   The default settings are as follows:

   - The maximum interval is 5 seconds.
   - The minimum interval is 50 milliseconds.
   - The incremental interval is 200 milliseconds.

Configuring the RIP packet sending rate

About this task

Perform this task to set the interval for sending RIP packets and the maximum number of RIP packets that can be sent at each interval. This feature can avoid excessive RIP packets from affecting system performance and consuming too much bandwidth.

Procedure

1. Enter system view.

   system-view

2. Configure the RIP packet sending rate.

   - Execute the following commands in sequence to configure the RIP packet sending rate for all interfaces:

```
rip [ process-id ] [ vpn-instance vpn-instance-name ]
output-delay time count count
```

By default, an interface sends up to three RIP packets every 20 milliseconds.

- Execute the following commands in sequence to configure the RIP packet sending rate for an interface:

```
interface interface-type interface-number
rip output-delay time count count
```

By default, the interface uses the RIP packet sending rate configured for the RIP process that the interface runs.

Setting the maximum length of RIP packets

About this task

The packet length of RIP packets determines how many routes can be carried in a RIP packet. Set the maximum length of RIP packets to make good use of link bandwidth.

When authentication is enabled, follow these guidelines to ensure packet forwarding:

- For simple authentication, the maximum length of RIP packets must be no less than 52 bytes.
- For MD5 authentication (with packet format defined in RFC 2453), the maximum length of RIP packets must be no less than 56 bytes.
- For MD5 authentication (with packet format defined in RFC 2082), the maximum length of RIP packets must be no less than 72 bytes.

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Set the maximum length of RIP packets.

```
rip max-packet-length value
```

By default, the maximum length of RIP packets is 512 bytes.

Configuring RIP network management

About this task

You can use network management software to manage the RIP process to which MIB is bound.

Procedure

1. Enter system view.

```
system-view
```

2. Bind MIB to a RIP process.

```
rip mib-binding process-id
```

By default, MIB is bound to the RIP process with the smallest process ID.

Configuring RIP GR

About this task

GR ensures forwarding continuity when a routing protocol restarts or an active/standby switchover occurs.

Two routers are required to complete a GR process. The following are router roles in a GR process:

- **GR restarter**—Graceful restarting router. It must have GR capability.
- **GR helper**—A neighbor of the GR restarter. It helps the GR restarter to complete the GR process.

After RIP restarts on a router, the router must learn RIP routes again and update its FIB table, which causes network disconnections and route reconvergence.

With the GR feature, the restarting router (known as the GR restarter) can notify the event to its GR capable neighbors. GR capable neighbors (known as GR helpers) maintain their adjacencies with the router within a GR interval. During this process, the FIB table of the router does not change. After the restart, the router contacts its neighbors to retrieve its FIB.

By default, a RIP-enabled device acts as the GR helper. Perform this task on the GR restarter.

Restrictions and guidelines

You cannot enable RIP NSR on a device that acts as GR restarter.

Procedure

1. Enter system view.
   `system-view`
2. Enter RIP view.
   `rip` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]
3. Enable GR for RIP.
   `graceful-restart`
   By default, RIP GR is disabled.
4. (Optional.) Set the GR interval.
   `graceful-restart interval` *`interval`*
   By default, the GR interval is 60 seconds.

Enabling RIP NSR

About this task

Nonstop Routing (NSR) allows the device to back up the routing information from the active RIP process to the standby RIP process. After an active/standby switchover, NSR can complete route regeneration without tearing down adjacencies or impacting forwarding services.

NSR does not require the cooperation of neighboring devices to recover routing information, and it is typically used more often than GR.

Restrictions and guidelines

A device that has RIP NSR enabled cannot act as GR restarter.

Procedure

1. Enter system view.
   `system-view`

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable RIP NSR.

   non-stop-routing

   By default, RIP NSR is disabled.

   RIP NSR enabled for a RIP process takes effect only on that process. As a best practice, enable RIP NSR for each process if multiple RIP processes exist.

Configuring BFD for RIP

About BFD for RIP

RIP detects route failures by periodically sending requests. If it receives no response for a route within a certain time, RIP considers the route unreachable. To speed up convergence, perform this task to enable BFD for RIP. For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

RIP supports the following BFD detection modes:

- **Single-hop echo detection**—Detection mode for a directly connected neighbor. In this mode, a BFD session is established only when the directly connected neighbor has route information to send.
- **Single-hop echo detection for a specific destination**—Detection mode for a directly connected neighbor. In this mode, a BFD session is established to the specified RIP neighbor when RIP is enabled on the local interface. When BFD detects a unidirectional link, the local device will not receive or send any RIP packets through the interface to improve convergence speed. When the link recovers, the interface can send RIP packets again.
- **Bidirectional control detection**—Detection mode for both directly and indirectly connected neighbors. In this mode, a BFD session is established only when both ends have routes to send and BFD is enabled on the receiving interface.

Restrictions and guidelines

The **rip bfd enable** and **rip bfd enable destination** commands are mutually exclusive.

The **bfd all-interfaces enable** command in RIP view enables BFD for RIP on all interfaces of the RIP process. When link flapping occurs on one of the interfaces, you can disable BFD for RIP on that interface to avoid network instability. For this purpose, you must use the **rip bfd disable** command instead of the **undo rip bfd enable** command in interface view.

Configuring single-hop echo detection (for a directly connected RIP neighbor)

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

3. Enable BFD for RIP.

   - Execute the following commands in sequence to enable BFD on all interfaces of a RIP process:

rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

bfd all-interfaces enable

- Execute the following commands in sequence to enable BFD on an interface:

  interface *interface-type interface-number*

  rip bfd enable

By default, BFD for RIP is disabled.

Configuring single-hop echo detection (for a specific destination)

Restrictions and guidelines

This feature applies only to RIP neighbors that are directly connected.

Procedure

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, no source IP address is configured for BFD echo packets.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD for RIP.

   rip bfd enable destination *ip-address*

   By default, BFD for RIP is disabled.

Configuring bidirectional control detection for an indirectly connected neighbor

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify a RIP neighbor.

   peer *ip-address*

   By default, RIP does not unicast updates to any peer.

   Because the **undo peer** command does not remove the neighbor relationship immediately, executing the command cannot bring down the BFD session immediately.

4. Enable BFD for RIP.

   - Execute the following commands in sequence to enable BFD on all interfaces of a RIP process:

     bfd all-interfaces enable

   - Execute the following commands in sequence to enable BFD on an interface:

     interface *interface-type interface-number*

     rip bfd enable

By default, BFD for RIP is disabled.

Configuring bidirectional control detection for a directly connected neighbor

1. Enter system view.

   system-view

2. Enable BFD for RIP.
   - Execute the following commands in sequence to enable BFD on all interfaces of a RIP process:

     rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

     bfd all-interfaces enable ctrl

   - Execute the following commands in sequence to enable BFD on an interface:

     interface *interface-type interface-number*

     rip bfd enable ctrl

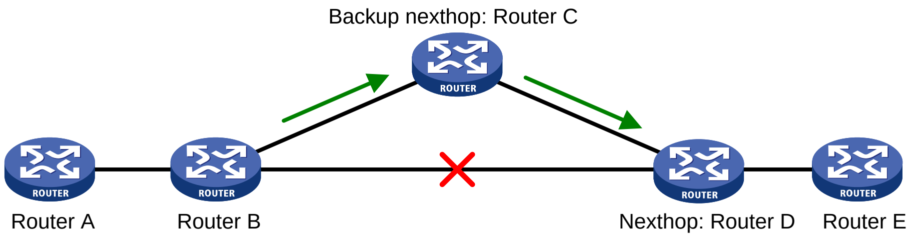
   By default, BFD for RIP is disabled.

Configuring RIP FRR

About RIP FRR

A link or router failure on a path can cause packet loss and even routing loop until RIP completes routing convergence based on the new network topology. FRR enables fast rerouting to minimize the impact of link or node failures.

Figure 1 Network diagram for RIP FRR



As shown in Figure 1, configure FRR on Router B by using a routing policy to specify a backup next hop. When the primary link fails, RIP directs packets to the backup next hop. At the same time, RIP calculates the shortest path based on the new network topology, and forwards packets over that path after network convergence.

Restrictions and guidelines for RIP FRR

RIP FRR takes effect only for RIP routes learned from directly connected neighbors.

RIP FRR is available only when the state of primary link (with Layer 3 interfaces staying up) changes from bidirectional to unidirectional or down.

Equal-cost routes do not support RIP FRR.

Enabling RIP FRR

1. Enter system view.

   system-view

2. Configure a routing policy for FRR.

   You must specify a next hop by using the **apply fast-reroute backup-interface** command in the routing policy.

   For more information about routing policy configuration, see "Configuring routing policies."

3. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

4. Enable RIP FRR.

   fast-reroute route-policy *route-policy-name*

   By default, RIP FRR is disabled.

Enabling BFD bidirectional control detection for RIP FRR

About this task

By default, RIP FRR does not use BFD to detect primary link failures. For quicker RIP FRR, use BFD on the primary link of redundant links to detect link failure.

Restrictions and guidelines

You must configure bidirectional control detection on both ends of a link for it to take effect.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD bidirectional control detection for RIP FRR.

   rip primary-path-detect bfd ctrl

   By default, BFD bidirectional control detection for RIP FRR is disabled.

Enabling BFD single-hop echo detection for RIP FRR

About this task

By default, RIP FRR does not use BFD to detect primary link failures. For quicker RIP FRR, use BFD single-hop echo detection on the primary link of redundant links to detect link failure.

Procedure

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interfaces.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD single-hop echo detection for RIP FRR.

   rip primary-path-detect bfd echo

   By default, BFD single-hop echo detection for RIP FRR is disabled.

Enhancing RIP security

Enabling zero field check for incoming RIPv1 messages

About this task

Some fields in the RIPv1 message must be set to zero. These fields are called "zero fields." You can enable zero field check for incoming RIPv1 messages. If a zero field of a message contains a non-zero value, RIP does not process the message. If you are certain that all messages are trustworthy, disable zero field check to save CPU resources.

This feature does not apply to RIPv2 packets, because they have no zero fields.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable zero field check for incoming RIPv1 messages.

   checkzero

   By default, zero field check is disabled for incoming RIPv1 messages.

Enabling source IP address check for incoming RIP updates

About this task

Perform this task to enable source IP address check for incoming RIP updates.

- Upon receiving a message on an Ethernet interface, RIP compares the source IP address of the message with the IP address of the interface. If they are not in the same network segment, RIP discards the message.
- Upon receiving a message on a PPP interface, RIP checks whether the source address of the message is the IP address of the peer interface. If not, RIP discards the message.

Procedure

1. Enter system view.

   system-view

2. Enter RIP view.

   rip [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable source IP address check for incoming RIP messages.

   validate-source-address

   By default, source IP address check is disabled for incoming RIP updates.

Configuring RIPv2 message authentication

About this task

Perform this task to enable authentication on RIPv2 messages.

RIPv2 supports simple authentication, MD5 authentication, and keychain authentication. For more information about keychains, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure RIPv2 authentication.

   `rip authentication-mode` { **`keychain`** *`keychain-name`* { **`rfc2453`** | **`rfc4822`** } | **`md5`** { **`rfc2082`** { **`cipher`** | **`plain`** } *`string key-id`* | **`rfc2453`** { **`cipher`** | **`plain`** } *`string`* } | **`simple`** { **`cipher`** | **`plain`** } *`string`* }

   By default, RIPv2 authentication is not configured.

   RIPv1 does not support authentication. Although you can specify an authentication mode for RIPv1 in interface view, the configuration does not take effect.

Display and maintenance commands for RIP

Execute **`display`** commands in any view and execute **`reset`** commands in user view.

Task	Command
Display RIP current status and configuration information.	**`display rip`** [*`process-id`*]
Display RIP GR information.	**`display rip`** [*`process-id`*] **`graceful-restart`**
Display RIP NSR information.	**`display rip`** [*`process-id`*] **`non-stop-routing`**
Display active routes in the RIP database.	**`display rip`** *`process-id`* **`database`** [*`ip-address`* { *`mask-length`* | *`mask`* }]
Display RIP interface information.	**`display rip`** *`process-id`* **`interface`** [*`interface-type interface-number`*]
Display neighbor information for a RIP process.	**`display rip`** *`process-id`* **`neighbor`** [*`interface-type interface-number`*]
Display routing information for a RIP process.	**`display rip`** *`process-id`* **`route`** [*`ip-address`* { *`mask-length`* | *`mask`* } [**`verbose`**] | **`peer`** *`ip-address`* | **`statistics`**]
Reset a RIP process.	**`reset rip`** *`process-id`* **`process`**
Clear the statistics for a RIP process.	**`reset rip`** *`process-id`* **`statistics`**

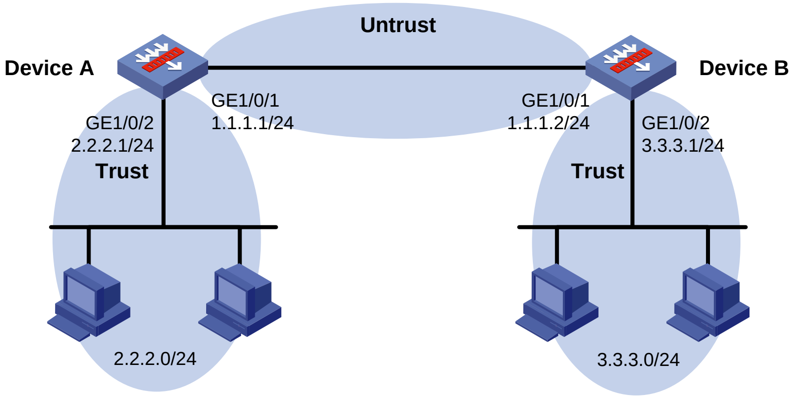
RIP configuration examples

Example: Configuring basic RIP

Network configuration

As shown in Figure 2, Device A and Device B are attached to networks 2.2.2.0/24 and 3.3.3.0/24, respectively. Configure RIPv2 on Device A and Device B to enable communication between the two networks.

Figure 2 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Trust] quit
   ```

3. Configure security policies:

   a. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones, and enable the devices to establish RIP neighbor relationship with each other.

      # Configure a rule named **riplocalin** to allow Device A to receive RIP packets from Device B.

      ```
      [DeviceA] security-policy ip
      [DeviceA-security-policy-ip] rule name riplocalin
      [DeviceA-security-policy-ip-0-riplocalin] source-zone untrust
      [DeviceA-security-policy-ip-0-riplocalin] destination-zone local
      [DeviceA-security-policy-ip-0-riplocalin] service rip
      ```

```
[DeviceA-security-policy-ip-0-riplocalin] action pass
[DeviceA-security-policy-ip-0-riplocalin] quit
```

# Configure a rule named **riplocalout** to allow Device A to send RIP packets to Device B.

```
[DeviceA-security-policy-ip] rule name riplocalout
[DeviceA-security-policy-ip-1-riplocalout] source-zone local
[DeviceA-security-policy-ip-1-riplocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-riplocalout] service rip
[DeviceA-security-policy-ip-1-riplocalout] action pass
[DeviceA-security-policy-ip-1-riplocalout] quit
```

b. Configure a security policy to permit traffic between the **Trust** and **Untrust** security zones, and enable communication between network 2.2.2.0/24 and network 3.3.3.0/24.

# Configure a rule named **trust-untrust** to permit packets from network 2.2.2.0/24 to network 3.3.3.0/24.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-2-trust-untrust] destination-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-2-trust-untrust] action pass
[DeviceA-security-policy-ip-2-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit packets from network 3.3.3.0/24 to network 2.2.2.0/24.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-3-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-3-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-3-untrust-trust] source-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-3-trust-untrust] destination-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-3-untrust-trust] action pass
[DeviceA-security-policy-ip-3-untrust-trust] quit
[DeviceA-security-policy-ip-3] quit
```

4. Configure basic RIP functions.

```
[DeviceA] rip
[DeviceA-rip-1] network 1.1.1.0
[DeviceA-rip-1] network 2.2.2.0
[DeviceA-rip-1] version 2
[DeviceA-rip-1] undo summary
[DeviceA-rip-1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
```

```
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies:
   a. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones, and enable the devices to establish RIP neighbor relationship with each other.

      # Configure a rule named **riplocalin** to allow Device B to receive RIP packets from Device A.

      ```
      [DeviceB] security-policy ip
      [DeviceB-security-policy-ip] rule name riplocalin
      [DeviceB-security-policy-ip-0-riplocalin] source-zone untrust
      [DeviceB-security-policy-ip-0-riplocalin] destination-zone local
      [DeviceB-security-policy-ip-0-riplocalin] service rip
      [DeviceB-security-policy-ip-0-riplocalin] action pass
      [DeviceB-security-policy-ip-0-riplocalin] quit
      ```

      # Configure a rule named **riplocalout** to allow Device B to send RIP packets to Device A.

      ```
      [DeviceB-security-policy-ip] rule name riplocalout
      [DeviceB-security-policy-ip-1-riplocalout] source-zone local
      [DeviceB-security-policy-ip-1-riplocalout] destination-zone untrust
      [DeviceB-security-policy-ip-1-riplocalout] service rip
      [DeviceB-security-policy-ip-1-riplocalout] action pass
      [DeviceB-security-policy-ip-1-riplocalout] quit
      ```

   b. Configure a security policy to permit traffic between the **Trust** and **Untrust** security zones, and enable communication between network 3.3.3.0/24 and network 2.2.2.0/24.

      # Configure a rule named **trust-untrust** to permit packets from network 3.3.3.0/24 to network 2.2.2.0/24.

      ```
      [DeviceB-security-policy-ip] rule name trust-untrust
      [DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
      [DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 3.3.3.0 24
      [DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 2.2.2.0 24
      [DeviceB-security-policy-ip-2-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit packets from network 2.2.2.0/24 to network 3.3.3.0/24.

      ```
      [DeviceB-security-policy-ip] rule name untrust-trust
      [DeviceB-security-policy-ip-3-untrust-trust] source-zone untrust
      [DeviceB-security-policy-ip-3-untrust-trust] destination-zone trust
      [DeviceB-security-policy-ip-3-untrust-trust] source-ip-subnet 2.2.2.0 24
      [DeviceB-security-policy-ip-3-untrust-trust] destination-ip-subnet 3.3.3.0 24
      [DeviceB-security-policy-ip-3-untrust-trust] action pass
      [DeviceB-security-policy-ip-3-untrust-trust] quit
      [DeviceB-security-policy-ip-3] quit
      ```

4. Configure basic RIP functions.

   ```
   [DeviceB] rip
   [DeviceB-rip-1] version 2
   [DeviceB-rip-1] undo summary
   [DeviceB-rip-1] quit
   ```

```
    [DeviceB] interface gigabitethernet 1/0/1
    [DeviceB-GigabitEthernet1/0/1] rip 1 enable
    [DeviceB-GigabitEthernet1/0/1] quit
    [DeviceB] interface gigabitethernet 1/0/2
    [DeviceB-GigabitEthernet1/0/2] rip 1 enable
    [DeviceB-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Display the RIP routing table of Device A.

```
[Device A] display rip 1 route
 Route Flags: R - RIP
              P - Permanent, A - Aging, S - Suppressed, G - Garbage-collect
              D - Direct, O - Optimal, F - Flush to RIB
----------------------------------------------------------------------------
 Peer 1.1.1.2 on GigabitEthernet1/0/1
      Destination/Mask        Nexthop           Cost    Tag     Flags   Sec
      3.3.3.0/24              1.1.1.1           1       0       RAOF    10
 Local route
      Destination/Mask        Nexthop           Cost    Tag     Flags   Sec
      2.2.2.0/24              0.0.0.0           0       0       RDOF    -
      1.1.1.0/24              0.0.0.0           0       0       RDOF    -
```

The output shows that the hosts in network 2.2.2.0/24 can communicate with the hosts in network 3.3.3.0/24.

Contents

Configuring RIPng

About RIPng

RIP next generation (RIPng), as an extension of RIP-2 for support of IPv6, is a distance vector routing protocol. It employs UDP to exchange route information through port 521. Most RIP concepts are applicable to RIPng.

RIPng routing metrics

RIPng uses a hop count to measure the distance to a destination. The hop count is the metric or cost. The hop count from a router to a directly connected network is 0. The hop count between two directly connected routers is 1. When the hop count is greater than or equal to 16, the destination network or host is unreachable.

RIPng route entries

RIPng stores route entries in a database. Each route entry contains the following elements:

- **Destination address**—IPv6 address of a destination host or a network.
- **Next hop address**—IPv6 address of the next hop.
- **Egress interface**—Egress interface of the route.
- **Metric**—Cost from the local router to the destination.
- **Route time**—Time elapsed since the most recent update. The time is reset to 0 every time the route entry is updated.
- **Route tag**—Used for route control. For more information, see "Configuring routing policies."

RIPng packets and advertisement

RIPng multicasts request and response packets to exchange routing information. It uses FF02::9 as the destination address and link-local address FE80::/10 as the source address. RIPng exchanges routing information as follows:

1. When RIPng starts or needs to update some route entries, it sends a multicast request packet to neighbors.
2. When a RIPng neighbor receives the request packet, it sends back a response packet that contains the local routing table. RIPng can also advertise route updates in response packets periodically or advertise a triggered update caused by a route change.
3. After RIPng receives the response, it checks the validity of the response before adding routes to its routing table, including the following details:
    - Whether the source IPv6 address is the link-local address.
    - Whether the port number is correct.
4. A response packet that fails the check is discarded.

Protocols and standards

- RFC 2080, *RIPng for IPv6*
- RFC 2081, *RIPng Protocol Applicability Statement*

RIPng tasks at a glance

To configure RIPng, perform the following tasks:

1. Configuring basic RIPng
2. (Optional.) Configuring RIPng route control
   - Configuring an additional routing metric
   - Configuring RIPng route summarization
   - Advertising a default route
   - Configuring received/redistributed route filtering
   - Setting a preference for RIPng
   - Configuring RIPng route redistribution
3. (Optional.) Tuning and optimizing the RIPng network
   - Setting RIPng timers
   - Configuring split horizon and poison reverse
   - Setting the maximum number of ECMP routes
   - Configuring the RIPng packet sending rate
   - Setting the interval for sending triggered updates
4. (Optional.) Enhancing RIPng availability
   - Configuring RIPng GR
   - Configuring RIPng NSR
   - Configuring RIPng FRR
5. (Optional.) Enhancing RIPng security
   - Configuring zero field check for RIPng packets
   - Applying an IPsec profile

Configuring basic RIPng

1. Enter system view.

   ```
   system-view
   ```

2. Enable RIPng and enter its view.

   ```
   ripng [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

   By default, RIPng is disabled.

3. Return to system view.

   ```
   quit
   ```

4. Enter interface view.

   ```
   interface interface-type interface-number
   ```

5. Enable RIPng on the interface.

   ```
   ripng process-id enable
   ```

   By default, RIPng is disabled on the interface.

   If RIPng is not enabled on an interface, the interface does not send or receive any RIPng route.

Configuring RIPng route control

Configuring an additional routing metric

About this task

An additional routing metric (hop count) can be added to the metric of an inbound or outbound RIPng route.

- An outbound additional metric is added to the metric of a sent route, and it does not change the route's metric in the routing table.
- An inbound additional metric is added to the metric of a received route before the route is added into the routing table, and the route's metric is changed.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify an inbound additional routing metric.

   ripng metricin *value*

   The default additional metric of an inbound route is 0.

4. Specify an outbound additional routing metric.

   ripng metricout *value*

   The default additional metric of an outbound route is 1.

Configuring RIPng route summarization

About this task

RIPng route summarization is interface-based. RIPng advertises a summary route based on the longest match.

RIPng route summarization improves network scalability, reduces routing table size, and increases routing table lookup efficiency.

RIPng advertises a summary route with the smallest metric of all the specific routes.

For example, RIPng has two specific routes to be advertised through an interface: 1:11:11::24 with a metric of a 2 and 1:11:12::34 with a metric of 3. Configure route summarization on the interface, so RIPng advertises a single route 11::0/16 with a metric of 2.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Advertise a summary IPv6 prefix.

   ripng summary-address *ipv6-address prefix-length*

   By default, no summary IPv6 prefix is configured on the interface.

Advertising a default route

About this task

You can configure RIPng to advertise a default route with the specified cost to its neighbors.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure RIPng to advertise a default route.

   ripng default-route { **only** | **originate** } [**cost** *cost-value* | **route-policy** *route-policy-name*] *

   By default, RIPng does not advertise a default route.

   This command advertises a default route on the current interface regardless of whether the default route exists in the local IPv6 routing table.

Configuring received/redistributed route filtering

About this task

Perform this task to filter received or redistributed routes by using an IPv6 ACL or IPv6 prefix list. You can also configure RIPng to filter routes redistributed from other routing protocols and routes from a specified neighbor.

Procedure

1. Enter system view.

   system-view

2. Enter RIPng view.

   ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Configure a filter policy to filter received routes.

   filter-policy { *ipv6-acl-number* | **prefix-list** *prefix-list-name* } **import**

   By default, RIPng does not filter received routes.

4. Configure a filter policy to filter redistributed routes.

   filter-policy { *ipv6-acl-number* | **prefix-list** *prefix-list-name* } **export** [*protocol* [*process-id*]]

   By default, RIPng does not filter redistributed routes.

Setting a preference for RIPng

About this task

Routing protocols each have a preference. When they find routes to the same destination, the route found by the routing protocol with the highest preference is selected as the optimal route. You can manually set a preference for RIPng. The smaller the value, the higher the preference.

Procedure

1. Enter system view.

   system-view

2. Enter RIPng view.

   ```
   ripng [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Set a preference for RIPng.

   ```
   preference { preference | route-policy route-policy-name } *
   ```

   By default, the preference of RIPng is 100.

Configuring RIPng route redistribution

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIPng view.

   ```
   ripng [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Configure route redistribution.
   - Redistribute routes from BGP4+.

     ```
     import-route bgp4+ [ as-number ] [ allow-ibgp ] [ cost cost-value |
     route-policy route-policy-name ] *
     ```

   - Redistribute direct or static routes.

     ```
     import-route { direct | static } [ cost cost-value | route-policy
     route-policy-name ] *
     ```

   - Redistribute routes from IPv6 IS-IS, OSPFv3, or other RIPng processes.

     ```
     import-route { isisv6 | ospfv3 | ripng } [ process-id ] [ allow-direct
     | cost cost-value | route-policy route-policy-name ] *
     ```

   By default, RIPng does not redistribute routes.

4. (Optional.) Set a default routing metric for redistributed routes.

   ```
   default cost cost-value
   ```

   The default metric of redistributed routes is 0.

Tuning and optimizing the RIPng network

Setting RIPng timers

About this task

You can adjust RIPng timers to optimize the performance of the RIPng network.

Restrictions and guidelines

When you adjust RIPng timers, consider the network performance, and perform unified configurations on routers running RIPng to avoid unnecessary network traffic or route oscillation.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIPng view.

   ```
   ripng [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Set RIPng timers.

   ```
   timers { garbage-collect garbage-collect-value | suppress
   suppress-value | timeout timeout-value | update update-value } *
   ```

The default settings are as follows:

- The update timer is 30 seconds.
- The timeout timer is 180 seconds.
- The suppress timer is 120 seconds.
- The garbage-collect timer is 120 seconds.

Configuring split horizon and poison reverse

Restrictions and guidelines for split horizon and poison reverse

When you configure split horizon and poison reverse, following these restrictions and guidelines:

- If both split horizon and poison reverse are configured, only the poison reverse feature takes effect.
- Split horizon disables RIPng from sending routes through the interface where the routes were learned to prevent routing loops between neighbors. As a best practice, enable split horizon to prevent routing loops in normal cases.
- Poison reverse enables a route learned from an interface to be advertised through the interface. However, the metric of the route is set to 16, which means the route is unreachable.

Configuring split horizon

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable split horizon.

   `ripng split-horizon`

   By default, split horizon is enabled.

Configuring poison reverse

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable poison reverse.

   `ripng poison-reverse`

   By default, poison reverse is disabled.

Setting the maximum number of ECMP routes

1. Enter system view.

   `system-view`

2. Enter RIPng view.

   `ripng` `[` *`process-id`* `] [` **`vpn-instance`** *`vpn-instance-name`* `]`

3. Set the maximum number of ECMP routes.

   `maximum load-balancing` *`number`*

Configuring the RIPng packet sending rate

About this task

Perform this task to specify the interval for sending RIPng packets and the maximum number of RIPng packets that can be sent at each interval. This feature can avoid excessive RIPng packets from affecting system performance and consuming too much bandwidth.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIPng view.

   ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Configuring the RIPng packet sending rate.

   - Execute the following commands in sequence to configure the RIPng packet sending rate in RIPng view:

     ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

     output-delay *time* **count** *count*

     By default, an interface that runs the RIPng process sends a maximum of three RIPng packets every 20 milliseconds.

   - Execute the following commands in sequence to configure the RIPng packet sending rate in interface view:

     interface *interface-type interface-number*

     ripng output-delay *time* **count** *count*

     By default, an interface uses the RIPng packet sending rate of the RIPng process that it runs.

Setting the interval for sending triggered updates

About this task

Perform this task to avoid network overhead and reduce system resource consumption caused by frequent RIPng triggered updates.

You can use the **timer triggered** command to set the maximum interval, minimum interval, and incremental interval for sending RIPng triggered updates.

For a stable network, the minimum interval is used. If network changes become frequent, the triggered update sending interval is incremented by the incremental interval × 2^{n-2} for each triggered update until the maximum interval is reached. The value *n* is the number of triggered update times.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIPng view.

   ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the interval for sending triggered updates.

   timer triggered *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   The default maximum interval is 5 seconds, the default minimum interval is 50 milliseconds, and the default incremental interval is 200 milliseconds.

Configuring RIPng GR

About this task

GR ensures forwarding continuity when a routing protocol restarts or an active/standby switchover occurs.

Two routers are required to complete a GR process. The following are router roles in a GR process:

- **GR restarter**—Graceful restarting router. It must have GR capability.
- **GR helper**—A neighbor of the GR restarter. It helps the GR restarter to complete the GR process.

After RIPng restarts on a router, the router must learn RIPng routes again and updates its FIB table, which causes network disconnections and route reconvergence.

With the GR feature, the restarting router (known as the GR restarter) can notify the event to its GR capable neighbors. GR capable neighbors (known as GR helpers) maintain their adjacencies with the router within a configurable GR interval. During this process, the FIB table of the router does not change. After the restart, the router contacts its neighbors to retrieve its FIB.

By default, a RIPng-enabled device acts as the GR helper. Perform this task on the GR restarter.

Restrictions and guidelines

You cannot enable RIPng NSR on a device that acts as GR restarter.

Procedure

1. Enter system view.

   `system-view`

2. Enable RIPng and enter RIPng view.

   `ripng` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

3. Enable the GR capability for RIPng.

   `graceful-restart`

   By default, RIPng GR is disabled.

4. (Optional.) Set the GR interval.

   `graceful-restart interval` *`interval`*

   The default GR interval is 60 seconds.

Configuring RIPng NSR

About this task

Nonstop routing (NSR) backs up RIPng routing information from the active process to the standby process. After an active/standby switchover, NSR can complete route regeneration without tearing down adjacencies or impacting forwarding services.

NSR does not require the cooperation of neighboring devices to recover routing information, and it is typically used more often than GR.

Restrictions and guidelines

RIPng NSR enabled for a RIPng process takes effect only on that process. If multiple RIPng processes exist, enable RIPng NSR for each process as a best practice.

A device that has RIPng NSR enabled cannot act as GR restarter.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIPng view.

   `ripng` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

3. Enable RIPng NSR.

   `non-stop-routing`

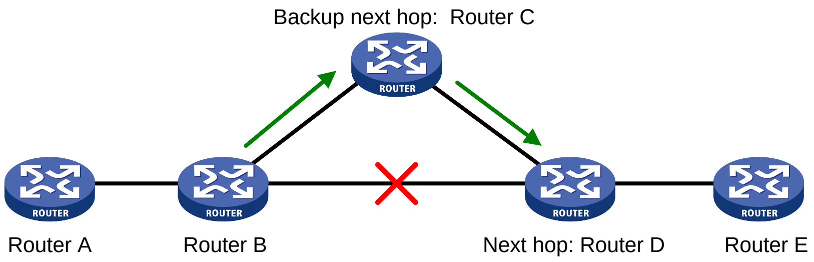
   By default, RIPng NSR is disabled.

Configuring RIPng FRR

About RIPng FRR

A link or router failure on a path can cause packet loss and even routing loop until RIPng completes routing convergence based on the new network topology. FRR enables fast rerouting to minimize the impact of link or node failures.

Figure 1 Network diagram for RIPng FRR



As shown in Figure 1, configure FRR on Router B by using a routing policy to specify a backup next hop. When the primary link fails, RIPng directs packets to the backup next hop. At the same time, RIPng calculates the shortest path based on the new network topology. Then, the device forwards packets over that path after network convergence.

Restrictions and guidelines for RIPng FRR

RIPng FRR is available only when the state of the primary link (with Layer 3 interfaces staying up) changes from bidirectional to unidirectional or down.

RIPng FRR is only effective for RIPng routes that are learned from directly connected neighbors.

Equal-cost routes do not support RIPng FRR.

Enabling RIPng FRR

1. Enter system view.

   `system-view`

2. Configure a routing policy.

   You must specify a next hop by using the **`apply ipv6 fast-reroute backup-interface`** command in a routing policy and specify the routing policy for FRR.

   For more information about routing policy configuration, see "Configuring routing policies."

3. Enter RIPng view.

ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

4. Enable RIPng FRR.

   fast-reroute route-policy *route-policy-name*

   By default, RIPng FRR is disabled.

Enabling BFD for RIPng FRR

About this task

By default, RIPng FRR does not use BFD to detect primary link failures. For quicker RIPng FRR, use BFD single-hop echo detection on the primary link of redundant links to detect link failure.

Procedure

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ipv6 *ipv6-address*

   By default, the source IP address of BFD echo packets is not configured.

   As a best practice, do not configure the source IP address on the same network segment as any local interfaces.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD single-hop echo detection for RIPng FRR.

   ripng primary-path-detect bfd echo

   By default, BFD single-hop echo detection is disabled for RIPng FRR.

Enhancing RIPng security

Configuring zero field check for RIPng packets

About this task

Some fields in the RIPng packet header must be zero. These fields are called zero fields. You can enable zero field check for incoming RIPng packets. If a zero field of a packet contains a non-zero value, RIPng does not process the packets. If you are certain that all packets are trustworthy, disable the zero field check to save CPU resources.

Procedure

1. Enter system view.

   system-view

2. Enter RIPng view.

   ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable the zero field check for incoming RIPng packets.

   checkzero

   By default, zero field check for incoming RIPng packets is enabled.

Applying an IPsec profile

About this task

To protect routing information and prevent attacks, you can configure RIPng to authenticate protocol packets by using an IPsec profile.

An IPsec profile contains inbound and outbound security parameter indexes (SPIs). RIPng compares the inbound SPI defined in the IPsec profile with the outbound SPI in the received packets. Two RIPng devices accept the packets from each other and establish a neighbor relationship only if the SPIs are the same and the relevant IPsec profiles match.

For more information about IPsec profiles, see *Security Configuration Guide*.

Restrictions and guidelines

You can apply an IPsec profile to a RIPng process or to an interface. If an interface and its process each have an IPsec profile, the IPsec profile applied to the interface takes effect.

Applying an IPsec profile to a process

1. Enter system view.
   system-view
2. Enter RIPng view.
   ripng [*process-id*] [**vpn-instance** *vpn-instance-name*]
3. Apply an IPsec profile to the process.
   enable ipsec-profile *profile-name*
   By default, no IPsec profile is applied.

Applying an IPsec profile to an interface

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Apply an IPsec profile to the interface.
   ripng ipsec-profile *profile-name*
   By default, no IPsec profile is applied.

Display and maintenance commands for RIPng

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display configuration information for a RIPng process.	**display ripng** [*process-id*]
Display RIPng GR information.	**display ripng** [*process-id*] **graceful-restart**
Display RIPng NSR information.	**display ripng** [*process-id*] **non-stop-routing**
Display routes in the RIPng database.	**display ripng** *process-id* **database** [*ipv6-address prefix-length*]
Display interface information for a RIPng	**display ripng** *process-id* **interface**

Task	Command
process.	[*interface-type interface-number*]
Display neighbor information for a RIPng process.	**display ripng** *process-id* **neighbor** [*interface-type interface-number*]
Display the routing information for a RIPng process.	**display ripng** *process-id* **route** [*ipv6-address prefix-length* [**verbose**] | **peer** *ipv6-address* | **statistics**]
Restart a RIPng process.	**reset ripng** *process-id* **process**
Clear statistics for a RIPng process.	**reset ripng** *process-id* **statistics**

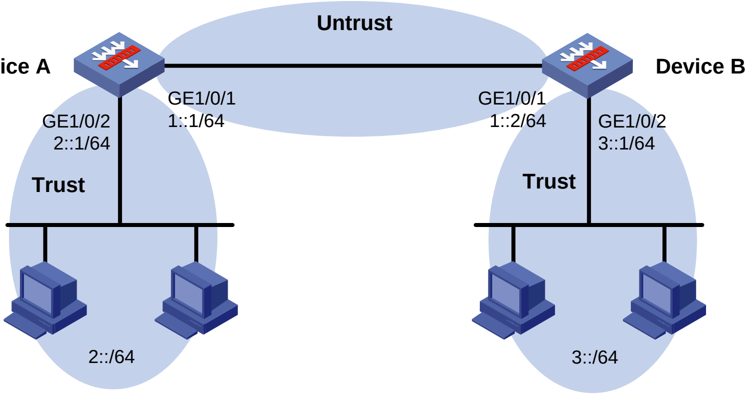
RIPng configuration examples

Example: Configuring basic RIPng

Network configuration

As shown in Figure 2, Device A and Device B are attached to networks 2::/64 and 3::/64, respectively. Configure RIPng on Device A and Device B to enable communication between the two networks.

Figure 2 Network diagram



Configuring Device A

1. Assign IPv6 addresses to interfaces:

   # Assign an IPv6 address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ipv6 address 1::1 64
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
   ```

```
[DeviceA-security-zone-Trust] quit
```

3. Configure security policies:
   a. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones, and enable the devices to establish RIPng neighbor relationship with each other.

      # Configure a rule named **ripnglocalin** to allow Device A to receive RIPng packets from Device B.

      ```
      [DeviceA] security-policy ipv6
      [DeviceA-security-policy-ipv6] rule name ripnglocalin
      [DeviceA-security-policy-ipv6-0-ripnglocalin] source-zone untrust
      [DeviceA-security-policy-ipv6-0-ripnglocalin] destination-zone local
      [DeviceA-security-policy-ipv6-0-ripnglocalin] service rip
      [DeviceA-security-policy-ipv6-0-ripnglocalin] action pass
      [DeviceA-security-policy-ipv6-0-ripnglocalin] quit
      ```

      # Configure a rule named **ripnglocalout** to allow Device A to send RIPng packets to Device B.

      ```
      [DeviceA-security-policy-ipv6] rule name ripnglocalout
      [DeviceA-security-policy-ipv6-1-ripnglocalout] source-zone local
      [DeviceA-security-policy-ipv6-1-ripnglocalout] destination-zone untrust
      [DeviceA-security-policy-ipv6-1-ripnglocalout] service rip
      [DeviceA-security-policy-ipv6-1-ripnglocalout] action pass
      [DeviceA-security-policy-ipv6-1-ripnglocalout] quit
      ```

   b. Configure a security policy to permit traffic between the **Trust** and **Untrust** security zones, and enable communication between network 2::/64 and network 3::/64.

      # Configure a rule named **trust-untrust** to permit packets from network 2::/64 to network 3::/64.

      ```
      [DeviceA-security-policy-ipv6] rule name trust-untrust
      [DeviceA-security-policy-ipv6-2-trust-untrust] source-zone trust
      [DeviceA-security-policy-ipv6-2-trust-untrust] destination-zone untrust
      [DeviceA-security-policy-ipv6-2-trust-untrust] source-ip-subnet 2:: 64
      [DeviceA-security-policy-ipv6-2-trust-untrust] destination-ip-subnet 3:: 64
      [DeviceA-security-policy-ipv6-2-trust-untrust] action pass
      [DeviceA-security-policy-ipv6-2-trust-untrust] quit
      ```

      # Configure a rule named **untrust-trust** to permit packets from network 3::/64 to network 2::/64.

      ```
      [DeviceA-security-policy-ipv6] rule name untrust-trust
      [DeviceA-security-policy-ipv6-3-untrust-trust] source-zone untrust
      [DeviceA-security-policy-ipv6-3-untrust-trust] destination-zone trust
      [DeviceA-security-policy-ipv6-3-untrust-trust] source-ip-subnet 3:: 64
      [DeviceA-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 2:: 64
      [DeviceA-security-policy-ipv6-3-untrust-trust] action pass
      [DeviceA-security-policy-ipv6-3-untrust-trust] quit
      [DeviceA-security-policy-ipv6] quit
      ```

4. Configure basic RIPng functions.

   ```
   [DeviceA] ripng 1
   [DeviceA-ripng-1] quit
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ripng 1 enable
   [DeviceA-GigabitEthernet1/0/1] quit
   [DeviceA] interface gigabitethernet 1/0/2
   ```

```
[DeviceA-GigabitEthernet1/0/2] ripng 1 enable
[DeviceA-GigabitEthernet1/0/2] quit
```

Configuring Device B

1. Assign IPv6 addresses to interfaces:

   # Assign an IPv6 address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ipv6 address 1::2 64
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IPv6 addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [DeviceB] security-zone name untrust
   [DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceB-security-zone-Untrust] quit
   [DeviceB] security-zone name trust
   [DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceB-security-zone-Trust] quit
   ```

3. Configure security policies:

   a. Configure a security policy to permit traffic between the **Untrust** and **Local** security zones, and enable the devices to establish RIPng neighbor relationship with each other.

      # Configure a rule named **ripnglocalin** to allow Device B to receive RIPng packets from Device A.

      ```
      [DeviceB] security-policy ipv6
      [DeviceB-security-policy-ipv6] rule name ripnglocalin
      [DeviceB-security-policy-ipv6-0-ripnglocalin] source-zone untrust
      [DeviceB-security-policy-ipv6-0-ripnglocalin] destination-zone local
      [DeviceB-security-policy-ipv6-0-ripnglocalin] service rip
      [DeviceB-security-policy-ipv6-0-ripnglocalin] action pass
      [DeviceB-security-policy-ipv6-0-ripnglocalin] quit
      ```

      # Configure a rule named **ripnglocalout** to allow Device B to send RIPng packets to Device A.

      ```
      [DeviceB-security-policy-ipv6] rule name ripnglocalout
      [DeviceB-security-policy-ipv6-1-ripnglocalout] source-zone local
      [DeviceB-security-policy-ipv6-1-ripnglocalout] destination-zone untrust
      [DeviceB-security-policy-ipv6-1-ripnglocalout] service rip
      [DeviceB-security-policy-ipv6-1-ripnglocalout] action pass
      [DeviceB-security-policy-ipv6-1-ripnglocalout] quit
      ```

   b. Configure a security policy to permit traffic between the **Trust** and **Untrust** security zones, and enable communication between network 2::/64 and network 3::/64.

      # Configure a rule named **trust-untrust** to permit packets from network 3::/64 to network 2::/64.

      ```
      [DeviceB-security-policy-ipv6] rule name trust-untrust
      [DeviceB-security-policy-ipv6-2-trust-untrust] source-zone trust
      [DeviceB-security-policy-ipv6-2-trust-untrust] destination-zone untrust
      [DeviceB-security-policy-ipv6-2-trust-untrust] source-ip-subnet 3:: 64
      [DeviceB-security-policy-ipv6-2-trust-untrust] destination-ip-subnet 2:: 64
      [DeviceB-security-policy-ipv6-2-trust-untrust] action pass
      [DeviceB-security-policy-ipv6-2-trust-untrust] quit
      ```

# Configure a rule named **untrust-trust** to permit packets from network 2::/64 to network 3::/64.

```
[DeviceB-security-policy-ipv6] rule name untrust-trust
[DeviceB-security-policy-ipv6-3-untrust-trust] source-zone untrust
[DeviceB-security-policy-ipv6-3-untrust-trust] destination-zone trust
[DeviceB-security-policy-ipv6-3-untrust-trust] source-ip-subnet 2:: 64
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 3:: 64
[DeviceB-security-policy-ipv6-3-untrust-trust] action pass
[DeviceB-security-policy-ipv6-3-untrust-trust] quit
[DeviceB-security-policy-ipv6] quit
```

4. Configure basic RIPng functions.

```
[DeviceB] ripng 1
[DeviceB-ripng-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ripng 1 enable
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ripng 1 enable
[DeviceB-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Display the RIPng routing table of Device A.

```
[Device A] display ripng 1 route
 Route Flags: A - Aging, S - Suppressed, G - Garbage-collect, D - Direct
               O - Optimal, F - Flush to RIB
----------------------------------------------------------------------------
 Peer FE80::76A5::A3FF:FE7B:205 on GigabitEthernet1/0/1
 Destination 3::/64,
     via FE80::76A5:A3FF:FE7B:205, cost 1, tag 0, AOF, 17 secs
 Local route
 Destination 2::/64,
     via ::, cost 0, tag 0, DOF
 Destination 1::/64,
     via ::, cost 0, tag 0, DOF
```

The output shows that the hosts in network 2::/64 can communicate with the hosts in network 3::/64.

Contents

Configuring OSPF

About OSPF

Open Shortest Path First (OSPF) is a link-state IGP developed by the OSPF working group of the IETF. OSPF version 2 is used for IPv4. OSPF refers to OSPFv2 throughout this chapter.

OSPF features

OSPF has the following features:

- **Wide scope**—Supports multiple network sizes and several hundred routers in an OSPF routing domain.
- **Fast convergence**—Advertises routing updates instantly upon network topology changes.
- **Loop free**—Computes routes with the SPF algorithm to avoid routing loops.
- **Area-based network partition**—Splits an AS into multiple areas to facilitate management. This feature reduces the LSDB size on routers to save memory and CPU resources, and reduces route updates transmitted between areas to save bandwidth.
- **ECMP routing**—Supports multiple equal-cost routes to a destination.
- **Routing hierarchy**—Supports a 4-level routing hierarchy that prioritizes routes into intra-area, inter-area, external Type-1, and external Type-2 routes.
- **Authentication**—Supports area- and interface-based packet authentication to ensure secure packet exchange.
- **Support for multicasting**—Multicasts protocol packets on some types of links to avoid impacting other devices.

OSPF packets

OSPF messages are carried directly over IP. The protocol number is 89.

OSPF uses the following packet types:

- **Hello**—Periodically sent to find and maintain neighbors, containing timer values, information about the DR, BDR, and known neighbors.
- **Database description (DD)**—Describes the digest of each LSA in the LSDB, exchanged between two routers for data synchronization.
- **Link state request (LSR)**—Requests needed LSAs from a neighbor. After exchanging the DD packets, the two routers know which LSAs of the neighbor are missing from their LSDBs. They then exchange LSR packets requesting the missing LSAs. LSR packets contain the digest of the missing LSAs.
- **Link state update (LSU)**—Transmits the requested LSAs to the neighbor.
- **Link state acknowledgment (LSAck)**—Acknowledges received LSU packets. It contains the headers of received LSAs (an LSAck packet can acknowledge multiple LSAs).

LSA types

OSPF advertises routing information in Link State Advertisements (LSAs). The following LSAs are commonly used:

- **Router LSA**—Type-1 LSA, originated by all routers and flooded throughout a single area only. This LSA describes the collected states of the router's interfaces to an area.
- **Network LSA**—Type-2 LSA, originated for broadcast and NBMA networks by the designated router, and flooded throughout a single area only. This LSA contains the list of routers connected to the network.
- **Network Summary LSA**—Type-3 LSA, originated by Area Border Routers (ABRs), and flooded throughout the LSA's associated area. Each summary-LSA describes a route to a destination outside the area, yet still inside the AS (an inter-area route).
- **ASBR Summary LSA**—Type-4 LSA, originated by ABRs and flooded throughout the LSA's associated area. Type 4 summary-LSAs describe routes to Autonomous System Boundary Router (ASBR).
- **AS External LSA**—Type-5 LSA, originated by ASBRs, and flooded throughout the AS (except stub and NSSA areas). Each AS-external-LSA describes a route to another AS.
- **NSSA LSA**—Type-7 LSA, as defined in RFC 1587, originated by ASBRs in NSSAs and flooded throughout a single NSSA. NSSA LSAs describe routes to other ASs.
- **Opaque LSA**—LSA for OSPF extensions. Its format consists of a standard LSA header and application specific information. The opaque LSA includes Type 9, Type 10, and Type 11. The Type 9 opaque LSA is flooded into the local subnet. Grace LSA, used by graceful restart, is Type 9 LSA. The Type 11 is flooded throughout the AS.

OSPF areas

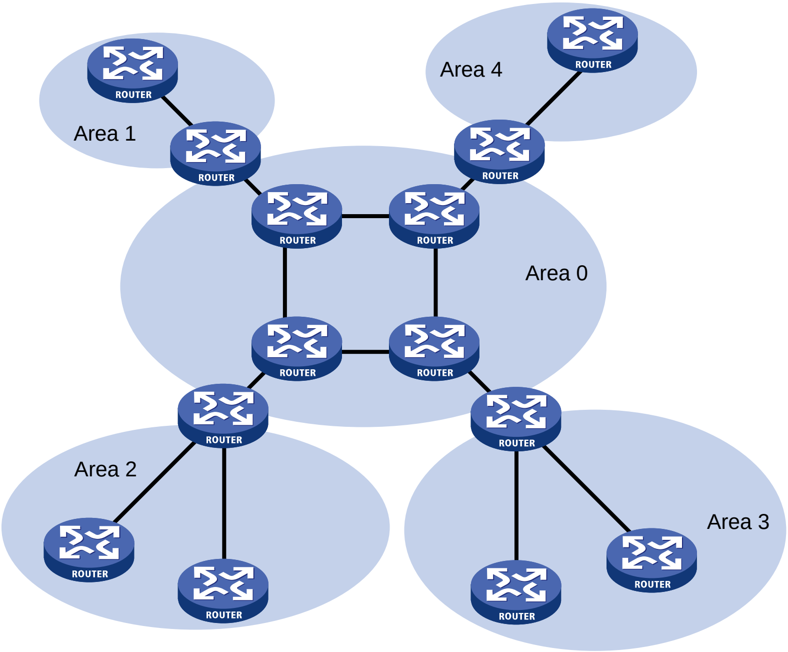
Area-based OSPF network partition

In large OSPF routing domains, SPF route computations consume too many storage and CPU resources, and enormous OSPF packets generated for route synchronization occupy excessive bandwidth.

To resolve these issues, OSPF splits an AS into multiple areas. Each area is identified by an area ID. The boundaries between areas are routers rather than links. A network segment (or a link) can only reside in one area as shown in Figure 1.

You can configure route summarization on ABRs to reduce the number of LSAs advertised to other areas and minimize the effect of topology changes.

Figure 1 Area-based OSPF network partition



Backbone area

Each AS has a backbone area that distributes routing information between non-backbone areas. Routing information between non-backbone areas must be forwarded by the backbone area. OSPF has the following requirements:

- All non-backbone areas must maintain connectivity to the backbone area.
- The backbone area must maintain connectivity within itself.

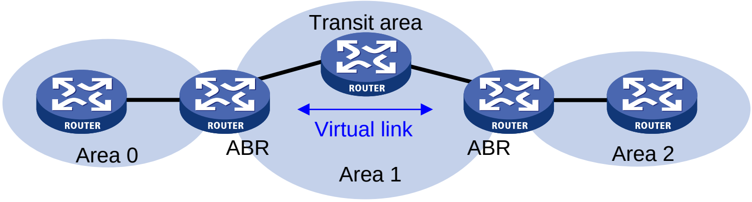
In practice, these requirements might not be met due to lack of physical links. OSPF virtual links can solve this issue.

Virtual links

A virtual link is established between two ABRs through a non-backbone area. It must be configured on both ABRs to take effect. The non-backbone area is called a transit area.

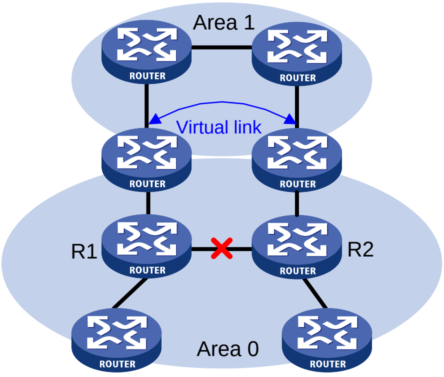
As shown in Figure 2, Area 2 has no direct physical link to the backbone Area 0. You can configure a virtual link between the two ABRs to connect Area 2 to the backbone area.

Figure 2 Virtual link application 1



Virtual links can also be used as redundant links. If a physical link failure breaks the internal connectivity of the backbone area, you can configure a virtual link to replace the failed physical link, as shown in Figure 3.

Figure 3 Virtual link application 2



The virtual link between the two ABRs acts as a point-to-point connection. You can configure interface parameters, such as hello interval, on the virtual link as they are configured on a physical interface.

The two ABRs on the virtual link unicast OSPF packets to each other, and the OSPF routers in between convey these OSPF packets as normal IP packets.

Stub area and totally stub area

A stub area does not distribute Type-5 LSAs to reduce the routing table size and LSAs advertised within the area. The ABR of the stub area advertises a default route in a Type-3 LSA so that the routers in the area can reach external networks through the default route.

To further reduce the routing table size and advertised LSAs, you can configure the stub area as a totally stub area. The ABR of a totally stub area does not advertise inter-area routes or external routes. It advertises a default route in a Type-3 LSA so that the routers in the area can reach external networks through the default route.

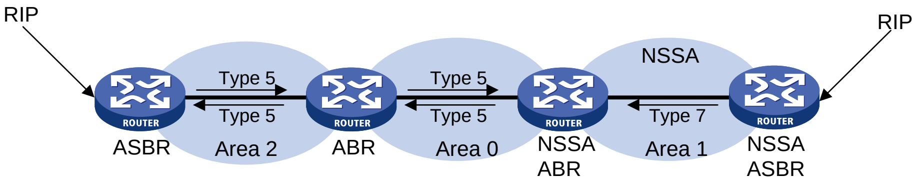
NSSA area and totally NSSA area

An NSSA area does not import AS external LSAs (Type-5 LSAs) but can import Type-7 LSAs generated by the NSSA ASBR. The NSSA ABR translates Type-7 LSAs into Type-5 LSAs and advertises the Type-5 LSAs to other areas.

As shown in Figure 4, the OSPF AS contains Area 1, Area 2, and Area 0. The other two ASs run RIP. Area 1 is an NSSA area where the ASBR redistributes RIP routes in Type-7 LSAs into Area 1. Upon receiving the Type-7 LSAs, the NSSA ABR translates them to Type-5 LSAs, and advertises the Type-5 LSAs to Area 0.

The ASBR of Area 2 redistributes RIP routes in Type-5 LSAs into the OSPF routing domain. However, Area 1 does not receive Type-5 LSAs because it is an NSSA area.

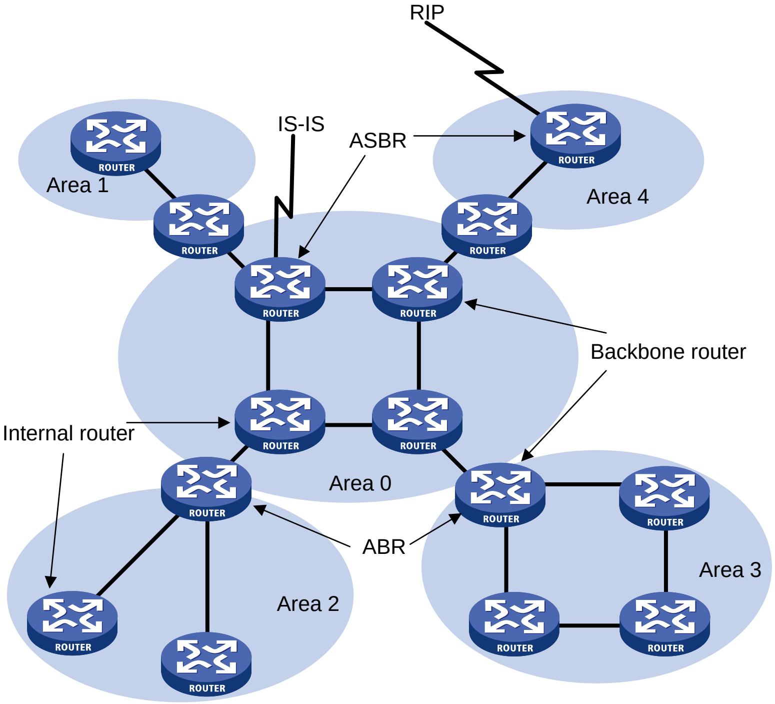
Figure 4 NSSA area



Router types

As shown in Figure 5, OSPF routers are classified into different types, including internal routers, ABRs, backbone routers, and ASBRs.

Figure 5 OSPF router types



Internal router

All interfaces on an internal router belong to one OSPF area.

ABR

An ABR belongs to more than two areas, one of which must be the backbone area. ABR connects the backbone area to a non-backbone area. An ABR and the backbone area can be connected through a physical or logical link.

Backbone router

No less than one interface of a backbone router must reside in the backbone area. All ABRs and internal routers in Area 0 are backbone routers.

ASBR

An ASBR exchanges routing information with another AS. An ASBR might not reside on the border of the AS. It can be an internal router or an ABR.

Route types

OSPF prioritizes routes into the following route levels:

- Intra-area route.
- Inter-area route.
- Type-1 external route.
- Type-2 external route.

The intra-area and inter-area routes describe the network topology of the AS. The external routes describe routes to external ASs.

A Type-1 external route has high credibility. The cost of a Type-1 external route = the cost from the router to the corresponding ASBR + the cost from the ASBR to the destination of the external route.

A Type-2 external route has low credibility. OSPF considers that the cost from the ASBR to the destination of a Type-2 external route is much greater than the cost from the ASBR to an OSPF internal router. The cost of a Type-2 external route = the cost from the ASBR to the destination of the Type-2 external route. If two Type-2 routes to the same destination have the same cost, OSPF takes the cost from the router to the ASBR into consideration to determine the best route.

Router ID

A router ID uniquely identifies a router in an AS. For a router to run OSPF, it must have a router ID. You can choose to manually specify a router ID or use the global router ID for an OSPF process.

Manual configuration

When you create an OSPF process, you can manually specify a router ID. To make sure the router ID is unique in the AS, you can specify the IP address of an interface on the router as the router ID.

Autoconfiguration

When you create an OSPF process, you can enable the OSPF process to automatically obtain a router ID. The OSPF process obtains a router ID in the following ways:

- During the startup of the OSPF process, the primary IPv4 address of the first interface that runs the process is specified as the router ID.
- During the reboot of the router, the primary IPv4 address of the first interface that runs the process is specified as the router ID.
- During the restart of the OSPF process, the highest primary IPv4 address of the loopback interface that runs the process is specified as the router ID. If no loopback address is available, the highest primary IPv4 address of the interface that runs the process is used, regardless of the interface state (up or down).

Using the global router ID

If you do not specify a router ID when creating an OSPF process, the global router ID is used. As a best practice, manually specify a router ID or enable the OSPF process to automatically obtain a router ID when you create the OSPF process.

Route calculation

OSPF computes routes in an area as follows:

- Each router generates LSAs based on the network topology around itself, and sends them to other routers in update packets.
- Each OSPF router collects LSAs from other routers to compose an LSDB. An LSA describes the network topology around a router, and the LSDB describes the entire network topology of the area.
- Each router transforms the LSDB to a weighted directed graph that shows the topology of the area. All the routers within the area have the same graph.
- Each router uses the SPF algorithm to compute a shortest path tree that shows the routes to the nodes in the area. The router itself is the root of the tree.

OSPF network types

OSPF classifies networks into the following types, depending on different link layer protocols:

- **Broadcast**—If the link layer protocol is Ethernet or FDDI, OSPF considers the network type as broadcast by default. On a broadcast network, hello, LSU, and LSAck packets are multicast to 224.0.0.5 that identifies all OSPF routers or to 224.0.0.6 that identifies the DR and BDR. DD packets and LSR packets are unicast.

- **NBMA**—If the link layer protocol is Frame Relay, ATM, or X.25, OSPF considers the network type as NBMA by default. OSPF packets are unicast on an NBMA network.
- **P2MP**—No link is P2MP type by default. P2MP must be a conversion from other network types such as NBMA. On a P2MP network, OSPF packets are multicast to 224.0.0.5.
- **P2P**—If the link layer protocol is PPP or HDLC, OSPF considers the network type as P2P. On a P2P network, OSPF packets are multicast to 224.0.0.5.

The following are the differences between NBMA and P2MP networks:

- NBMA networks are fully meshed. P2MP networks are not required to be fully meshed.
- NBMA networks require DR and BDR election. P2MP networks do not have DR or BDR.
- On an NBMA network, OSPF packets are unicast, and neighbors are manually configured. On a P2MP network, OSPF packets are multicast by default, and you can configure OSPF to unicast protocol packets.

DR and BDR

DR and BDR mechanism

On a broadcast or NBMA network, any two routers must establish an adjacency to exchange routing information with each other. If n routers are present on the network, n(n-1)/2 adjacencies are established. Any topology change on the network results in an increase in traffic for route synchronization, which consumes a large amount of system and bandwidth resources.

Using the DR and BDR mechanisms can solve this problem.

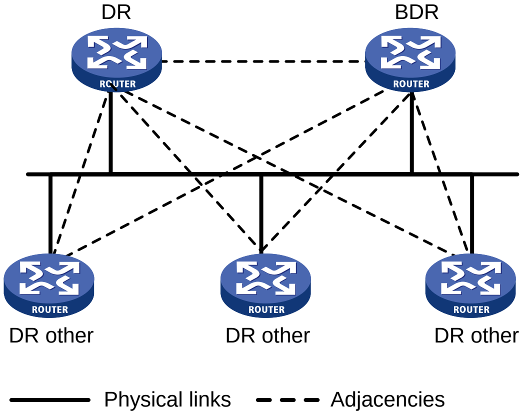
- **DR**—Elected to advertise routing information among other routers. If the DR fails, routers on the network must elect another DR and synchronize information with the new DR. Using this mechanism without BDR is time-consuming and is prone to route calculation errors.
- **BDR**—Elected along with the DR to establish adjacencies with all other routers. If the DR fails, the BDR immediately becomes the new DR, and other routers elect a new BDR.

Routers other than the DR and BDR are called DR Others. They do not establish adjacencies with one another, so the number of adjacencies is reduced.

The role of a router is subnet (or interface) specific. It might be a DR on one interface and a BDR or DR Other on another interface.

As shown in Figure 6, solid lines are Ethernet physical links, and dashed lines represent OSPF adjacencies. With the DR and BDR, only seven adjacencies are established.

Figure 6 DR and BDR in a network



NOTE:

In OSPF, neighbor and adjacency are different concepts. After startup, OSPF sends a hello packet on each OSPF interface. A receiving router checks parameters in the packet. If the parameters match its own, the receiving router considers the sending router an OSPF neighbor. Two OSPF neighbors establish an adjacency relationship after they synchronize their LSDBs through exchange of DD packets and LSAs.

DR and BDR election

DR election is performed on broadcast or NBMA networks but not on P2P and P2MP networks.

Routers in a broadcast or NBMA network elect the DR and BDR by router priority and ID. Routers with a router priority value higher than 0 are candidates for DR and BDR election.

The election votes are hello packets. Each router sends the DR elected by itself in a hello packet to all the other routers. If two routers on the network declare themselves as the DR, the router with the higher router priority wins. If router priorities are the same, the router with the higher router ID wins.

If a router with a higher router priority becomes active after DR and BDR election, the router cannot replace the DR or BDR until a new election is performed. Therefore, the DR of a network might not be the router with the highest priority, and the BDR might not be the router with the second highest priority.

OSPF state machines

Interface state machine

On receipt of link state information, OSPF establishes adjacency relationships with neighboring devices, and then exchanges LSAs with them. The state of an OSPF interface indicates the role of the device in an OSPF link. Two neighboring devices can correctly form an adjacency relationship by checking the other's interface state.

The interface state machine involves the following interface states:

- **Down**—Initial interface state.

  An OSPF interface in this state cannot be used for traffic sending or receiving.

- **Loopback**—The network-facing interface is looped back on the device.

  A loopback interface is not available for regular data transmission. However, to obtain link quality information on the interface, you might either send ICMP ping packets to the loopback interface or have a bit error test. To meet this potential requirement, each OSPF device advertises loopback interface information in Router LSAs.

- **Waiting**—The device is determining the DR and BDR for the network.

  When the device does not run for the DR or BDR, the interface starts a wait timer. The hello packets sent by the device does not contain any DR or BDR information unless the wait timer expires. The device cannot be elected as the DR or BDR on the network until it exits from **Waiting** state. The wait timer mechanism prevents unnecessary DR and BDR changes.

  This interface state is supported only on NBMA networks and broadcast networks.

- **Point-to-point**—The interface is connected to a physical P2P network or to a virtual link.

  If an interface enters this state, it attempts to establish an adjacency relationship with the neighboring device.

- **DR Other**—The interface is connected to a broadcast or NBMA network on which another device has been elected as the DR.

  A device in **DR Other** state is neither elected as the DR nor the BDR. When DR and BDR election is complete on the network, each DR Other device establishes adjacencies with both the DR and the BDR.

- **BDR**—The device is the BDR on the attached network.

As the BDR, the device establishes adjacencies with all other devices on the attached network. It will immediately become the new DR if the existing DR fails.

- **DR**—The device is the DR on the attached network.

  As the DR, the device establishes adjacencies with all other devices on the attached network.

A number of input events (IE) can cause OSPF interface state changes. As shown in Figure 7, the relationships between IEs and state changes form the OSPF interface state machine.

Figure 7 OSPF interface state machine

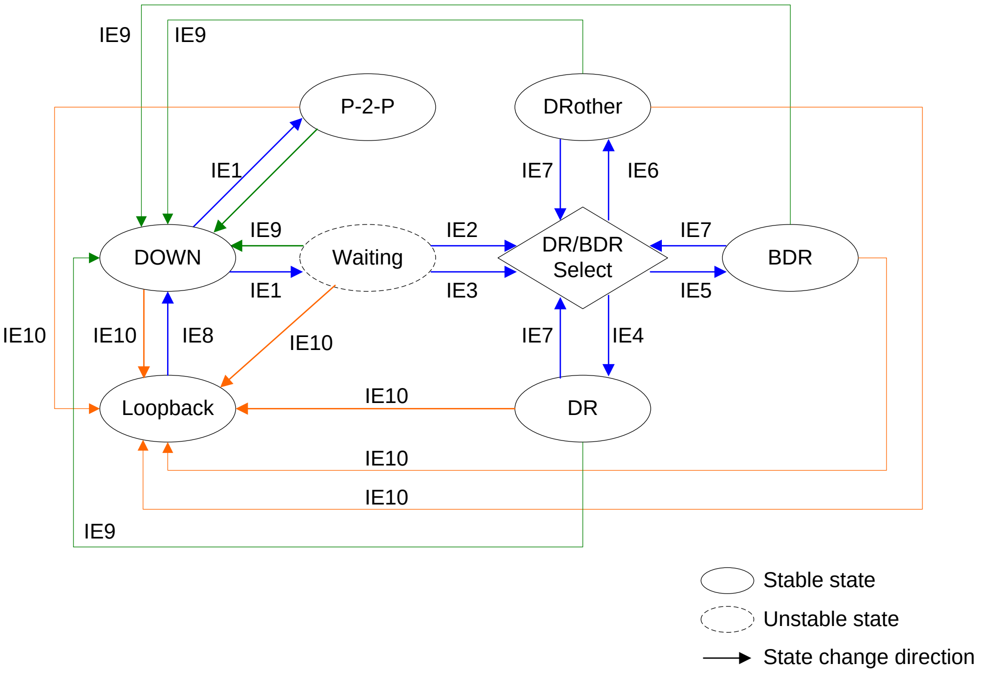


Table 1 shows the IEs that can cause interface state changes.

Table 1 Input events

Input event	Description
IE1	InterfaceUp event: Lower-level protocols have indicated that the interface is operational.
IE2	WaitTimer event: The wait timer has expired and the device can run for the DR or the BDR.
IE3	BackupSeen event: The device has detected the existence or non-existence of a BDR on the attached network through one of the following methods: • The interface receives a hello packet from a neighbor that claims itself to be the BDR. • The interface receives a hello packet from a neighbor that claims itself to be the DR. The hello packet also indicates that there is no BDR. This event marks an end to **Waiting** state and indicates that the device and the neighbor can have bidirectional communication.
IE4	The interface is elected as the DR.
IE5	The interface is elected as the BDR.
IE6	The interface is neither elected as the DR nor the BDR.
IE7	NeighborChange event: A neighbor associated with the interface has a change. The DR and BDR need to be re-elected. The following neighbor changes might lead to the re-election of DR and BDR:

	• The device can have bidirectional communication with a neighbor. • The device no longer has bidirectional communication with a neighbor. • A neighbor claims itself as DR or BDR. The device detects this event again by examining the hello packets from the neighbor. • A neighbor no longer claims itself as DR or BDR. The device detects this event again by examining the hello packets from the neighbor. • The DR priority for a neighbor has changed. The device detects this event again by examining the hello packets from the neighbor.
IE8	UnloopInd event: Network management or lower-level protocols has indicated that the interface is no longer looped back.
IE9	InterfaceDown event: Lower-level protocols has indicated that the interface is not operational. If this event occurs, the interface state might change to Down.
IE10	LoopInd event: Network management or lower-level protocols has indicated that the interface is looped back. If this event occurs, the interface state might change to Loopback.

Neighbor state machine

The process of OSPF adjacency establishment involves a series of neighbor state changes that transition from **Down** state to **Full** state.

The neighbor state machine involves the following neighbor states:

- **Down**—Intial state of a neighbor conversation.

  This state indicates that the device does not received any hello packets from the neighbor within the neighbor dead interval. An OSPF device sends hello packets at intervals of PollInterval.

  Whether an OSPF device sends hello packets to neighbors in **Down** state depends on its network type. The device sends hello packets to neighbors in **Down** state only when its network type is NBMA.
- **Attempt**—In this state, the device attempts to establish neighbor relationships with the maually specified neighbors by sending hello packets at intervals of HelloInterval.

  This state is valid only for the neighbors on NBMA networks.
- **Init**—This state indicates that the OSPF device has received a hello packet from a neighbor before the neighbor dead interval elapses. However, the OSPF device has not established bidirectional communication with the neighbor, because the received hello packet does not contain the OSPF device's router ID.
- **2-Way**—This state indicates that the OSPF device has established a neighbor relationship with a neighboring device. Each of the two devices has received a hello packet from another and has seen its own router ID in the received hello packet.

  If the neighbor relationship between two devices does not prompt to an adjacency, the neighbot state will remain 2-Way.

  DR and BDR election starts only when the neighbor state is 2-Way or greater.
- **ExStart**—This state indicates that the OSPF device and the neighbor is deciding which is the master. This step is to determine the sequence number for initial DD packets. The devices can then exchange DD packets in order.
- **Exhange**—This state indicates that the OSPF device is exchanging DD packets with the neighbor. The device describes its LSDB information in DD packets and sends them to the neighbor.
- **Loading**—This state indicates that the OSPF device and the neighbor are having bidirectional LSDB synchronization. They send LSRs to each other to request the most recent LSAs, and then synchronize their LSDBs through exchanges of LSUs and LSAcks.
- **Full**—This state indicates that the OSPF device and the neighbor has completed bidirectional LSDB synchronization and they are fully adjacent.

As shown in Figure 8, the relationships between IEs and state changes form the OSPF neighbor state machine.

Figure 8 OSPF neighbor state machine

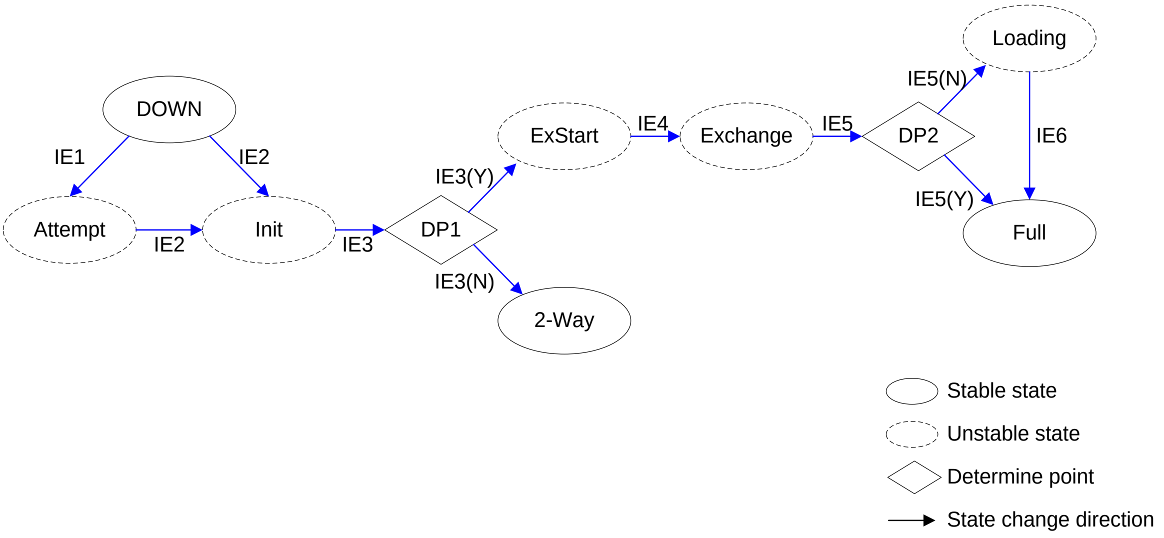


Table 2 shows the IEs that can cause neighbor state changes.

Table 2 Input events

Input event	Description
IE1	Start event: The OSPF device attempts to establish a neighbor relationship with the neighbor by sending hello packets at intervals of HelloInterval. This event is supported only on NBMA networks.
IE2	HelloReceived event: The OSPF device has received a hello packet from the neighbor.
IE3	2-WayReceived event: The OSPF device has received a hello packet from the neighbor and has seen its router ID in the received hello packet. The two neighboring devices have established bidirectional communication. The OSPF device then performs the following task: • If the OSPF device needs to form an adjacency with the neighbor, it changes the neighbor state to ExStart. • If the OSPF device should not form an adjacency with the neighbor, it changes the neighbor state to 2-Way.
IE4	NegotiationDone event: The OSPF device and the neighbor have completed master/secondary relationship negotiation and have determined the sequence number for initial DD packets.
IE5	ExchangeDone event: The OSPF device have exchangd DD packets with the neighbor. The OSPF device then performs the following task: • If the LSR list is empty, the device changes the neighbor state to Full. This neighbor state indicates that bidirectional LSDB sychronization is complete and the device has established an adjacency with the neighbor. • If the LSR list is not empty, the device changes the neighbor state to Loading, and then sends LSRs to the neighbor to request missing link state information. If the OSPF device should not form an adjacency with the neighbor, it changes the neighbor state to 2-Way.
IE6	LoadingDone event: The LSR list is already empty.

Protocols and standards

- RFC 1245, *OSPF protocol analysis*
- RFC 1246, *Experience with the OSPF protocol*
- RFC 1370, *Applicability Statement for OSPF*
- RFC 1403, *BGP OSPF Interaction*
- RFC 1745, *BGP4/IDRP for IP---OSPF Interaction*
- RFC 1765, *OSPF Database Overflow*
- RFC 1793, *Extending OSPF to Support Demand Circuits*
- RFC 2154, *OSPF with Digital Signatures*
- RFC 2328, *OSPF Version 2*
- RFC 3101, *OSPF Not-So-Stubby Area (NSSA) Option*
- RFC 3166, *Request to Move RFC 1403 to Historic Status*
- RFC 3509, *Alternative Implementations of OSPF Area Border Routers*
- RFC 4167, *Graceful OSPF Restart Implementation Report*
- RFC 4750, *OSPF Version 2 Management Information Base*
- RFC 4811, *OSPF Out-of-Band LSDB Resynchronization*
- RFC 4812, *OSPF Restart Signaling*
- RFC 5088, *OSPF Protocol Extensions for Path Computation Element (PCE) Discovery*
- RFC 5250, *The OSPF Opaque LSA Option*
- RFC 5613, *OSPF Link-Local Signaling*
- RFC 5642, *Dynamic Hostname Exchange Mechanism for OSPF*
- RFC 5709, *OSPFv2 HMAC-SHA Cryptographic Authentication*
- RFC 5786, *Advertising a Router's Local Addresses in OSPF Traffic Engineering (TE) Extensions*
- RFC 6571, *Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks*
- RFC 6860, *Hiding Transit-Only Networks in OSPF*
- RFC 6987, *OSPF Stub Router Advertisement*

Restrictions and guidelines: OSPF configuration

To run OSPF, you must first enable OSPF on the router. Make a proper configuration plan to avoid incorrect settings that can result in route blocking and routing loops.

OSPF tasks at a glance

To configure OSPF, perform the following tasks:

1. Configuring basic OSPF functions
   - Enabling an OSPF process
   - Creating an OSPF area
   - Enabling OSPF
   - Configuring OSPF dynamic host name mappings
2. (Optional.) Configuring OSPF stub and NSSA areas

- Configuring a stub area
- Configuring an NSSA area
- Configuring a virtual link

3. (Optional.) Configuring OSPF network types
    - Configuring the broadcast network type for an interface
    - Configuring the NBMA network type for an interface
    - Configuring the P2MP network type for an interface
    - Configuring the P2P network type for an interface
4. (Optional.) Configuring OSPF route control
    - Configuring OSPF inter-area route summarization
    - Configuring redistributed route summarization
    - Configuring received OSPF route filtering
    - Configuring Type-3 LSA filtering
    - Setting an OSPF cost for an interface
    - Setting the maximum number of ECMP routes
    - Setting OSPF preference
    - Configuring discard routes for summary networks
    - Redistributing routes from another routing protocol
    - Redistributing a default route
    - Advertising a host route
    - Advertising OSPF link state information to BGP
5. (Optional.) Setting OSPF timers
    - Configuring OSPF packet timers
    - Setting LSA transmission delay
    - Setting SPF calculation interval
    - Setting the minimum LSA arrival interval
    - Setting the LSA generation interval
    - Setting OSPF exit overflow interval
6. (Optional.) Configuring OSPF packet parameters
    - Disabling interfaces from receiving and sending OSPF packets
    - Adding the interface MTU into DD packets
    - Setting the DSCP value for outgoing OSPF packets
    - Setting the LSU transmit rate
7. (Optional.) Controlling LSA generation, advertisement, and reception
    - Setting the maximum number of external LSAs in LSDB
    - Filtering outbound LSAs on an interface
    - Filtering LSAs for the specified neighbor
8. (Optional.) Accelerating OSPF convergence speed
    - Enabling OSPF ISPF
    - Configuring prefix suppression
    - Configuring prefix prioritization
    - Configuring OSPF PIC
9. (Optional.) Configuring advanced OSPF features
    - Configuring stub routers

- Enabling compatibility with RFC 1583
- Enabling OSPF to group ECMP routes

10. (Optional.) Enhancing OSPF availability
    - Configuring OSPF GR
    - Configuring OSPF NSR
    - Configuring BFD for OSPF
    - Configuring OSPF FRR
11. (Optional.) Configuring OSPF security features
    - Configuring OSPF authentication
    - Configuring GTSM for OSPF
12. (Optional.) Configuring OSPF logging and SNMP notifications
    - Logging neighbor state changes
    - Configuring the OSPF logging feature
    - Configuring OSPF network management

Configuring basic OSPF functions

Enabling an OSPF process

1. Enter system view.

   system-view

2. (Optional.) Configure a global router ID.

   router id *router-id*

   By default, no global router ID is configured.

   If no global router ID is configured, the highest loopback interface IP address, if any, is used as the router ID. If no loopback interface IP address is available, the highest physical interface IP address is used, regardless of the interface status (up or down).

3. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

   By default, OSPF is disabled.

4. (Optional.) Configure a description for the OSPF process.

   description *text*

   By default, no description is configured for the OSPF process.

   As a best practice, configure a description for each OSPF process.

Creating an OSPF area

1. Enter system view.

   system-view

2. (Optional.) Configure a global router ID.

   router id *router-id*

   By default, no global router ID is configured.

3. Enter OSPF view.

ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

By default, OSPF is disabled.

4. (Optional.) Configure a description for the OSPF process.

   description *text*

   By default, no description is configured for the OSPF process.

   As a best practice, configure a description for each OSPF process.

5. Create an OSPF area and enter OSPF area view.

   area *area-id*

6. (Optional.) Configure a description for the area.

   description *text*

   By default, no description is configured for the area.

   As a best practice, configure a description for each OSPF area.

Enabling OSPF

About this task

To enable OSPF on a router, you must perform the following tasks:

1. Create an OSPF process.
2. Create an OSPF area for the process.
3. Specify a network in the area.

The interface attached to the network will run the OSPF process in the area. OSPF advertises direct routes of the interface.

OSPF supports multiple processes. To run multiple OSPF processes, you must specify an ID for each process. The process IDs take effect locally and has no influence on packet exchange between routers. Two routers with different process IDs can exchange packets.

Restrictions and guidelines for enabling OSPF

When you configure OSPF on an interface, follow these restrictions and guidelines:

- You can enable OSPF on the network where the interface resides or directly enable OSPF on that interface. If you configure both, the latter takes precedence.
- If the specified OSPF process and area do not exist, the operation creates an OSPF process and area for the interface. Disabling an OSPF process on an interface does not delete the OSPF process or the area.

Enabling OSPF on a network

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Create an OSPF area and enter OSPF area view.

   area *area-id*

4. Specify a network to enable the interface attached to the network to run the OSPF process in the area.

   network *ip-address wildcard-mask*

   By default, no network is specified to enable OSPF on the interface attached to the network.

A network can be added to only one area.

Enabling OSPF on an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable an OSPF process on the interface.

   ospf *process-id* **area** *area-id* [**exclude-subip**]

   By default, OSPF is disabled on an interface.

Configuring OSPF dynamic host name mappings

About this task

OSPF uses a router ID to uniquely identify a router in an AS. The length of router IDs is fixed at 4 bytes. It is inconvenient for the network administrator to memorize the router IDs in dotted decimal notation when verifying the OSPF neighbor relationships, routing tables, and LSDBs.

This task allows you to map a router ID to a host name. The mapping table is maintained by OSPF routers. Host names are more straightforward than router IDs in network maintenance, management, and failure diagnosis.

Restrictions and guidelines

OSPF uses Type-11 LSAs to carry information about the dynamic host name attribute. Therefore, make sure the opaque LSA reception and advertisement capability is enabled.

Procedure

1. Enter system view.

   system-view

2. Enable OSPF, and enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable the opaque LSA reception and advertisement capability.

   opaque-capability enable

   By default, the opaque LSA reception and advertisement capability is enabled.

4. Enable the OSPF dynamic host name mapping feature.

   hostname [*host-name*]

   By default, the OSPF dynamic host name mapping feature is disabled.

Configuring OSPF stub and NSSA areas

About OSPF stub and NSSA area configuration

This task allows you to configure an OSPF area as a stub area or NSSA area. It also allows you to create a virtual link if no connectivity can be achieved between a non-backbone area and backbone area, or in the backbone area.

Configuring a stub area

About this task

You can configure a non-backbone area at an AS edge as a stub area. To do so, execute the **stub** command on all routers attached to the area. The routing table size is reduced because Type-5 LSAs will not be flooded within the stub area. The ABR generates a default route into the stub area so all packets destined outside of the AS are sent through the default route.

To further reduce the routing table size and routing information exchanged in the stub area, configure a totally stub area by using the **stub no-summary** command on the ABR. AS external routes and inter-area routes will not be distributed into the area. All the packets destined for outside of the AS or area will be sent to the ABR for forwarding.

A stub or totally stub area cannot have an ASBR because external routes cannot be distributed into the area.

Restrictions and guidelines

Do not configure the backbone area as a stub area or totally stub area.

To configure an area as a stub area, execute the **stub** command on all routers attached to the area.

To configure an area as a totally stub area, execute the **stub** command on all routers attached to the area, and execute the **stub no-summary** command on the ABR.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enter area view.

   area *area-id*

4. Configure the area as a stub area.

   stub [**default-route-advertise-always** | **no-summary**] *

   By default, no stub area is configured.

5. (Optional.) Set a cost for the default route advertised to the stub area.

   default-cost *cost-value*

   By default, the cost for the default route advertised to the stub area is 1.

   This command takes effect only on the ABR of a stub area or totally stub area.

Configuring an NSSA area

About this task

A stub area cannot import external routes, but an NSSA area can import external routes into the OSPF routing domain while retaining other stub area characteristics.

To configure an area as a totally NSSA area, use the **nssa no-summary** command. The ABR of the area does not advertise inter-area routes into the area.

Restrictions and guidelines

Do not configure the backbone area as an NSSA area or totally NSSA area.

To configure an NSSA area, configure the **nssa** command on all the routers attached to the area.

To configure a totally NSSA area, configure the **nssa** command on all the routers attached to the area and configure the **nssa no-summary** command on the ABR.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enter area view.

   area *area-id*

4. Configure the area as an NSSA area.

   nssa [**default-route-advertise** [**cost** *cost-value* | **nssa-only** | **route-policy** *route-policy-name* | **type** *type*] * | **no-import-route** | **no-summary** | **suppress-fa** | [[[**translate-always**] [**translate-ignore-checking-backbone**]] | **translate-never**] | **translator-stability-interval** *value*] *

   By default, no area is configured as an NSSA area.

5. (Optional.) Set a cost for the default route advertised to the NSSA area.

   default-cost *cost-value*

   By default, the cost for the default route advertised to the NSSA area is 1.

   This command takes effect only on the ABR/ASBR on an NSSA area or totally NSSA area.

Configuring a virtual link

About this task

You can configure a virtual link to maintain connectivity between a non-backbone area and the backbone, or in the backbone itself.

Restrictions and guidelines

A virtual link cannot traverse a stub area, totally stub area, NSSA area, or totally NSSA area.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Create an OSPF area and enter OSPF area view.

   area *area-id*

4. Configure a virtual link.

   vlink-peer *router-id* [**dead** *seconds* | **hello** *seconds* | { { **hmac-md5** | **hmac-sha-256** | **md5** } *key-id* { **cipher** | **plain** } *string* | **keychain** *keychain-name* | **simple** { **cipher** | **plain** } *string* } | **retransmit** *seconds* | **trans-delay** *seconds*] *

   Configure this command on both ends of a virtual link. The **hello** and **dead** intervals must be identical on both ends of the virtual link.

Configuring OSPF network types

Based on the link layer protocol, OSPF classifies networks into different types, including broadcast, NBMA, P2MP, and P2P.

Restrictions and guidelines for configuring OSPF network types

When an NBMA network becomes fully meshed, change the network type to broadcast to avoid manual configuration of neighbors.

If any routers in a broadcast network do not support multicasting, change the network type to NBMA.

An NBMA network must be fully meshed. OSPF requires that an NBMA network be fully meshed. If a network is partially meshed, change the network type to P2MP.

If only two routers running OSPF exist on a network segment, you can change the network type to P2P to save costs.

Two broadcast-, NBMA-, and P2MP-interfaces can establish a neighbor relationship only when they are on the same network segment.

Configuring the broadcast network type for an interface

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the OSPF network type for the interface as broadcast.

   `ospf network-type broadcast`

   By default, the network type of an interface depends on the link layer protocol.

   When the link layer protocol is Ethernet or FDDI, OSPF classifies the network type as broadcast by default.

4. (Optional.) Set a router priority for the interface.

   `ospf dr-priority` *`priority`*

   The default router priority is 1.

Configuring the NBMA network type for an interface

Restrictions and guidelines

After you configure the network type as NBMA, you must specify neighbors and their router priorities because NBMA interfaces cannot find neighbors by broadcasting hello packets.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the OSPF network type for the interface as NBMA.

   `ospf network-type nbma`

By default, the network type of an interface varies by physical media.

4. (Optional.) Set a router priority for the interface.

   ospf dr-priority *priority*

   The default router priority for an interface is 1.

   The router priority configured with this command is for DR election.

5. Return to system view.

   quit

6. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

7. Specify an NBMA neighbor.

   peer *ip-address* [**dr-priority** *priority*]

   By default, no neighbor is specified.

   The priority configured with this command indicates whether a neighbor has the election right or not. If you configure the router priority for a neighbor as 0, the local router determines the neighbor has no election right. It does not send hello packets to this neighbor. However, if the local router is the DR or BDR, it still sends hello packets to the neighbor for neighbor relationship establishment.

Configuring the P2MP network type for an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the OSPF network type for the interface as P2MP.

   ospf network-type p2mp [**unicast**]

   By default, the network type of an interface depends on the link layer protocol.

   After you configure the OSPF network type for an interface as P2MP unicast, all packets are unicast over the interface. The interface cannot broadcast hello packets to discover neighbors, so you must manually specify the neighbors.

4. Return to system view.

   quit

5. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

6. Specify a P2MP neighbor.

   peer *ip-address* [**cost** *cost-value*]

   By default, no neighbor is specified

   This step is required if the interface network type is P2MP unicast.

Configuring the P2P network type for an interface

1. Enter system view.

   system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Configure the OSPF network type for the interface as P2P.

   ospf network-type p2p [**peer-address-check**]

   By default, the network type of an interface depends on the link layer protocol.

   When the link layer protocol is PPP, the interface network type is P2P by default.

Configuring OSPF route control

This section describes how to control the advertisement and reception of OSPF routing information, as well as route redistribution from other protocols.

Configuring OSPF inter-area route summarization

About this task

OSPF inter-area route summarization reduces the routing information exchanged between areas and the size of routing tables, and improves routing performance.

OSPF inter-area route summarization enables an ABR to summarize contiguous networks into a single network and advertise the network to other areas. For example, three internal networks 19.1.1.0/24, 19.1.2.0/24, and 19.1.3.0/24 are available within an area. You can configure the ABR to summarize the three networks into network 19.1.0.0/16, and advertise the summary network to other areas in a Type-3 LSA. This configuration reduces the scale of LSDBs on routers in other areas and the influence of topology changes.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Create an OSPF area and enter OSPF area view.

   area *area-id*

4. Configure ABR route summarization.

   abr-summary *ip-address* { *mask-length* | *mask* } [**advertise** | **not-advertise**] [**cost** *cost-value*]

   By default, route summarization is not configured on an ABR.

Configuring redistributed route summarization

About this task

Perform this task to enable an ASBR to summarize external routes within the specified address range into a single route. The ASBR advertises only Type-5 LSAs to reduce the number of LSAs in the LSDB.

An ASBR can summarize routes in the following LSAs:

- Type-5 LSAs.
- Type-7 LSAs in an NSSA area.

Restrictions and guidelines

If an ASBR (also an ABR) is a translator in an NSSA area, it summarizes routes in Type-5 LSAs translated from Type-7 LSAs. If it is not a translator, it does not summarize routes in in Type-5 LSAs translated from Type-7 LSAs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Configure ASBR route summarization.

   asbr-summary *ip-address* { *mask-length* | *mask* } [**cost** *cost-value* | **not-advertise** | **nssa-only** | **tag** *tag*] *

   By default, route summarization is not configured on an ASBR.

Configuring received OSPF route filtering

About this task

Perform this task to filter routes calculated using received LSAs.

The following filtering methods are available:

- Use an ACL or IP prefix list to filter routing information by destination address.
- Use the **gateway** *prefix-list-name* option to filter routing information by next hop.
- Use an ACL or IP prefix list to filter routing information by destination address. At the same time use the **gateway** *prefix-list-name* option to filter routing information by next hop.
- Use the **route-policy** *route-policy-name* option to filter routing information.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Configure OSPF to filter routes calculated using received LSAs.

   filter-policy { *ipv4-acl-number* [**gateway** *prefix-list-name*] | **gateway** *prefix-list-name* | **prefix-list** *prefix-list-name* [**gateway** *prefix-list-name*] | **route-policy** *route-policy-name* } **import**

   By default, OSPF accepts all routes calculated by using received LSAs.

Configuring Type-3 LSA filtering

About this task

Perform this task to filter Type-3 LSAs advertised into the local area or other areas on an ABR.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Create an OSPF area and enter OSPF area view.

   area *area-id*

4. Configure Type-3 LSA filtering.

   filter { *ipv4-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } { **export** | **import** }

   By default, the ABR does not filter Type-3 LSAs.

Setting an OSPF cost for an interface

About this task

Set an OSPF cost for an interface by using either of the following methods:

- Set the cost value in interface view.
- Set a bandwidth reference value for the interface. OSPF computes the cost with this formula: Interface OSPF cost = Bandwidth reference value (100 Mbps) / Expected interface bandwidth (Mbps). The expected bandwidth of an interface is configured with the **bandwidth** command (see *Interface Command Reference*).
  - If the calculated cost is greater than 65535, the value of 65535 is used. If the calculated cost is less than 1, the value of 1 is used.
  - If no cost or bandwidth reference value is configured for an interface, OSPF computes the interface cost based on the interface bandwidth and default bandwidth reference value.

Setting an OSPF cost for an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set an OSPF cost for the interface.

   ospf cost *cost-value*

   By default, the OSPF cost is calculated according to the interface bandwidth. For a loopback interface, the OSPF cost is 0 by default.

Setting a bandwidth reference value

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set a bandwidth reference value.

   bandwidth-reference *value*

   The default setting is 100 Mbps.

Setting the maximum number of ECMP routes

About this task

OSPF might find multiple optimal equal-cost routes to the same destination, which can be used to share the traffic load. This task allows you to set the maximum number of ECMP routes for OSPF.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the maximum number of ECMP routes.

   maximum load-balancing *number*

   By default, the maximum number of ECMP routes equals the maximum number of ECMP routes supported by the system.

Setting OSPF preference

About this task

A router can run multiple routing protocols, and each protocol is assigned a preference. If multiple routes are available to the same destination, the one with the highest protocol preference is selected as the best route.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set a preference for OSPF.

   preference [**ase**] { *preference* | **route-policy** *route-policy-name* } *

   By default, the preference of OSPF internal routes is 10 and the preference of OSPF external routes is 150.

Configuring discard routes for summary networks

About this task

Perform this task on an ABR or ASBR to specify whether to generate discard routes for summary networks. You can also specify a preference for the discard routes.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Configure discard routes for summary networks.

discard-route { **external** { *preference* | **suppression** } | **internal** { *preference* | **suppression** } } *

By default, the ABR or ASBR generates discard routes for summary networks and the default preference of discard routes is 255.

Redistributing routes from another routing protocol

About this task

On a router running OSPF and other routing protocols, you can configure OSPF to redistribute routes from other protocols. OSPF advertises the routes in Type-5 LSAs or Type-7 LSAs. In addition, you can configure OSPF to filter redistributed routes so that OSPF advertises only permitted routes.

Restrictions and guidelines

OSPF redistributes only active routes. To view route status information, use the **display ip routing-table protocol** command.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Configure route redistribution.

   import-route bgp [*as-number*] [**allow-ibgp**] [**cost** *cost-value* | **nssa-only** | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] *

   import-route { **direct** | **guard** | **static** } [**cost** *cost-value* | **nssa-only** | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] *

   import-route { **isis** | **ospf** | **rip** } [*process-id* | **all-processes**] [**allow-direct** | **cost** *cost-value* | **nssa-only** | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] *

   By default, OSPF does not redistribute routes.

   The **import-route bgp** command redistributes only EBGP routes. The **import-route bgp allow-ibgp** command redistributes both EBGP and IBGP routes, which might cause routing loops. Therefore, use it with caution. Support for the **allow-ibgp** keyword depends on the device model.

4. (Optional.) Configure OSPF to filter redistributed routes.

   filter-policy { *ipv4-acl-number* | **prefix-list** *prefix-list-name* } **export** [[**bgp** | **direct** | { **isis** | **ospf** | **rip** } [*process-id*] | **static**]

   By default, OSPF accepts all redistributed routes.

5. Configure the default parameters for redistributed routes (cost, tag, and type).

   default { **cost** *cost-value* | **tag** *tag* | **type** *type* } *

   By default, the cost is 1, the tag is 1, and the route type is 2

Redistributing a default route

About this task

The **import-route** command cannot redistribute a default external route. Perform this task to redistribute a default route.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Redistribute a default route.

   default-route-advertise [[**always** | **permit-calculate-other**] | **cost** *cost-value* | **route-policy** *route-policy-name* | **type** *type*] *

   default-route-advertise [**summary cost** *cost-value*]

   By default, no default route is redistributed.

   This command is applicable only to VPNs. The PE router advertises a default route in a Type-3 LSA to a CE router.

4. Configure the default parameters for redistributed routes (cost, tag, and type).

   default { **cost** *cost-value* | **tag** *tag* | **type** *type* } *

   By default, the cost is 1, the tag is 1, and the route type is 2

Advertising a host route

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enter area view.

   area *area-id*

4. Advertise a host route.

   host-advertise *ip-address cost-value*

   By default, OSPF does not advertise host routes that are not in the area.

Advertising OSPF link state information to BGP

About this task

After the device advertises OSPF link state information to BGP, BGP can advertise the information for intended applications. For more information about BGP LS, see "Configuring BGP."

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Advertise OSPF link state information to BGP.

   distribute bgp-ls [**instance-id** *id*] [**strict-link-checking**]

   By default, the device does not advertise OSPF link state information to BGP.

Setting OSPF timers

About setting OSPF timers

This task allows you to change OSPF packet timers to adjust the convergence speed and network load and tune the delay time for sending LSAs on low-speed links.

Configuring OSPF packet timers

About this task

An OSPF interface includes the following timers:

- **Hello timer**—Interval for sending hello packets. It must be identical on OSPF neighbors.
- **Poll timer**—Interval for sending hello packets to a neighbor that is down on the NBMA network.
- **Dead timer**—Interval within which if the interface does not receive any hello packet from the neighbor, it declares the neighbor is down.
- **LSA retransmission timer**—Interval within which if the interface does not receive any acknowledgment packets after sending an LSA to the neighbor, it retransmits the LSA.

Restrictions and guidelines

The default value for the hello interval and neighbor dead interval depends on the network type. When the network type for an interface is changed, the default hello interval and neighbor dead interval are restored. Make sure two neighboring interfaces are configured with the same hello interval and neighbor dead interval. Inconsistent settings will affect the OSPF neighbor relationship establishment.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the hello interval.

   ospf timer hello *seconds*

   The default hello interval on P2P and broadcast interfaces is 10 seconds. The default hello interval on P2MP and NBMA interfaces is 30 seconds.

4. Set the poll interval.

   ospf timer poll *seconds*

   The default setting is 120 seconds.

   The poll interval is a minimum of four times the hello interval.

5. Set the dead interval.

   ospf timer dead *seconds*

   The default dead interval on P2P and broadcast interfaces is 40 seconds. The default dead interval on P2MP and NBMA interfaces is 120 seconds.

   The dead interval must be a minimum of four times the hello interval on an interface.

6. Set the retransmission interval.

   ospf timer retransmit *interval*

   The default retransmission interval is 5 seconds.

A retransmission interval setting that is too small can cause unnecessary LSA retransmissions. Typically set a bigger interval than the round-trip time of a packet between two neighbors.

Setting LSA transmission delay

About this task

To avoid LSAs from aging out during transmission, set an LSA retransmission delay especially for low speed links.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Set the LSA transmission delay.

   `ospf trans-delay` *`seconds`*

   The default LSA transmission delay is 1 second.

Setting SPF calculation interval

About this task

LSDB changes result in SPF calculations. When the topology changes frequently, a large amount of network and router resources are occupied by SPF calculation. You can adjust the SPF calculation interval to reduce the impact.

For a stable network, the minimum interval is used. If network changes become frequent, the SPF calculation interval increases by the incremental interval × 2^{n-2} for each calculation until the maximum interval is reached. The value *n* is the number of calculation times.

Procedure

1. Enter system view.

   `system-view`

2. Enter OSPF view.

   `ospf` [*`process-id`* | **`router-id`** { **`auto-select`** | *`router-id`* } | **`vpn-instance`** *`vpn-instance-name`*] *

3. Set the SPF calculation interval.

   `spf-schedule-interval` { *`maximum-interval`* [*`minimum-interval`* [*`incremental-interval`*]] | **`millisecond`** *`interval`* }

   By default, the maximum interval is 5 seconds, the minimum interval is 50 milliseconds, and the incremental interval is 200 milliseconds.

Setting the minimum LSA arrival interval

About this task

OSPF drops any duplicate LSAs (with the same LSA type, LS ID, and router ID) within the minimum LSA arrival interval. This helps avoid overuse of bandwidth and router resources due to frequent network changes.

Procedure

1. Enter system view.

system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the minimum LSA arrival interval.

   lsa-arrival-interval *interval*

   By default, the minimum LSA arrival interval is 1000 milliseconds.

Setting the LSA generation interval

About this task

Adjust the LSA generation interval to protect network resources and routers from being overwhelmed by LSAs at the time of frequent network changes.

For a stable network, the minimum interval is used. If network changes become frequent, the LSA generation interval increases by the incremental interval × 2^{n-2} for each generation until the maximum interval is reached. The value *n* is the number of generation times.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the LSA generation interval.

   lsa-generation-interval *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   By default, the maximum interval is 5 seconds, the minimum interval is 50 milliseconds, and the incremental interval is 200 milliseconds.

Setting OSPF exit overflow interval

About this task

When the number of LSAs in the LSDB exceeds the upper limit, the LSDB is in an overflow state. In this state, OSPF does not receive any external LSAs and deletes the external LSAs generated by itself to save system resources.

This task allows you to configure the interval that OSPF exits overflow state.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the interval that OSPF exits overflow state.

   lsdb-overflow-interval *interval*

   By default, the OSPF exit overflow interval is 300 seconds. An interval of 0 means that OSPF does not exit overflow state.

Configuring OSPF packet parameters

Disabling interfaces from receiving and sending OSPF packets

About this task

To enhance OSPF adaptability and reduce resource consumption, you can set an OSPF interface to "silent." A silent OSPF interface blocks OSPF packets and cannot establish any OSPF neighbor relationship. However, other interfaces on the router can still advertise direct routes of the interface in Router LSAs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Disable interfaces from receiving and sending OSPF packets.

   silent-interface { *interface-type interface-number* | **all** }

   By default, an OSPF interface can receive and send OSPF packets.

   This command disables only the interfaces associated with the current process rather than other processes. Multiple OSPF processes can disable the same interface from receiving and sending OSPF packets.

Adding the interface MTU into DD packets

About this task

By default, an OSPF interface adds a value of 0 into the interface MTU field of a DD packet rather than the actual interface MTU. You can enable an interface to add its MTU into DD packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the interface to add its MTU into DD packets.

   ospf mtu-enable

   By default, the interface adds an MTU value of 0 into DD packets.

Setting the DSCP value for outgoing OSPF packets

About this task

The DSCP value specifies the precedence of outgoing packets.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the DSCP value for outgoing OSPF packets.

   dscp *dscp-value*

   By default, the DSCP value for outgoing OSPF packets is 48.

Setting the LSU transmit rate

About this task

During LSDB synchronization, if the local router has multiple neighbors, it must send many LSUs to each neighbor. When a neighbor receives excessive LSUs within a short time period, the following events might occur:

- The neighbor experiences degraded performance because it uses too many system resources to process the received LSU packets.
- The neighbor drops hello packets used for maintaining the neighbor relationship because it is busy dealing with the LSUs. As a result, the neighbor relationship is torn down. To reestablish a relationship to the neighbor, the local router must send more LSUs to the neighbor. This exacerbates the performance degradation.

This task allows you to limit the LSU transmit rate by setting the LSU transmit interval and the maximum number of LSUs that can be sent at each interval.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. (Optional.) Set the LSU transmit interval and the maximum number of LSUs that can be sent at each interval.

   transmit-pacing interval *interval* **count** *count*

   By default, an OSPF interface sends a maximum of three LSU packets every 20 milliseconds.

Controlling LSA generation, advertisement, and reception

Setting the maximum number of external LSAs in LSDB

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the maximum number of external LSAs in the LSDB.

   lsdb-overflow-limit *number*

   By default, the maximum number of external LSAs in the LSDB is not limited.

Filtering outbound LSAs on an interface

About this task

To reduce the LSDB size for the neighbor and save bandwidth, you can perform this task on an interface to filter LSAs to be sent to the neighbor.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Filter outbound LSAs on the interface.

   ospf database-filter { **all** | { **ase** [**acl** *ipv4-acl-number*] | **nssa** [**acl** *ipv4-acl-number*] | **summary** [**acl** *ipv4-acl-number*] } * }

   By default, the outbound LSAs are not filtered on the interface.

Filtering LSAs for the specified neighbor

About this task

On a P2MP network, a router might have multiple P2MP type OSPF neighbors. Perform this task to prevent the router from sending LSAs to the specified P2MP neighbor.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Filter LSAs for the specified P2MP neighbor.

   database-filter peer *ip-address* { **all** | { **ase** [**acl** *ipv4-acl-number*] | **nssa** [**acl** *ipv4-acl-number*] | **summary** [**acl** *ipv4-acl-number*] } * }

   By default, the LSAs for the specified P2MP neighbor are not filtered.

Accelerating OSPF convergence speed

Enabling OSPF ISPF

About this task

When the topology changes, Incremental Shortest Path First (ISPF) computes only the affected part of the SPT, instead of the entire SPT.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable OSPF ISPF.

   ispf enable

   By default, OSPF ISPF is enabled.

Configuring prefix suppression

About this task

By default, an OSPF interface advertises all of its prefixes in LSAs. To speed up OSPF convergence, you can suppress interfaces from advertising all of their prefixes. This feature helps improve network security by preventing IP routing to the suppressed networks.

When prefix suppression is enabled:

- On P2P and P2MP networks, OSPF does not advertise Type-3 links in Type-1 LSAs. Other routing information can still be advertised to ensure traffic forwarding.
- On broadcast and NBMA networks, the DR generates Type-2 LSAs with a mask length of 32 to suppress network routes. Other routing information can still be advertised to ensure traffic forwarding. If no neighbors exist, the DR does not advertise Type-3 links in Type-1 LSAs.

Restrictions and guidelines for prefix suppression

As a best practice, configure prefix suppression on all OSPF routers if you want to use prefix suppression.

Configuring prefix suppression for an OSPF process

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable prefix suppression for the OSPF process.

   prefix-suppression

   By default, prefix suppression is disabled for an OSPF process.

   This feature does not suppress the prefixes of secondary IP addresses, loopback interfaces, and passive interfaces.

Configuring prefix suppression for an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable prefix suppression for the interface.

   ospf prefix-suppression [**disable**]

   By default, prefix suppression is disabled on an interface.

   This feature does not suppress prefixes of secondary IP addresses.

Configuring prefix prioritization

About this task

This feature enables the device to install prefixes in descending priority order: critical, high, medium, and low. The prefix priorities are assigned through routing policies. When a route is assigned multiple prefix priorities, the route uses the highest priority.

By default, the 32-bit OSPF host routes have a medium priority and other routes have a low priority.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable prefix prioritization.

   prefix-priority route-policy *route-policy-name*

   By default, prefix prioritization is disabled.

Configuring OSPF PIC

About this task

Prefix Independent Convergence (PIC) enables the device to speed up network convergence by ignoring the number of prefixes.

Restrictions and guidelines for OSPF PIC

When both OSPF PIC and OSPF FRR are configured, OSPF FRR takes effect.

OSPF PIC applies only to inter-area routes and external routes.

Enabling OSPF PIC

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable PIC for OSPF.

   pic [**additional-path-always**]

   By default, OSPF PIC is enabled.

Configuring BFD control packet mode for OSPF PIC

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for OSPF PIC.

   ospf primary-path-detect bfd ctrl

   By default, BFD control packet mode is disabled for OSPF PIC.

   This mode requires BFD configuration on both OSPF routers on the link.

Configuring BFD echo packet mode for OSPF PIC

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interfaces.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD echo packet mode for OSPF PIC.

   ospf primary-path-detect bfd echo

   By default, BFD echo packet mode is disabled for OSPF PIC.

   This mode requires BFD configuration on one OSPF router on the link.

Configuring advanced OSPF features

Configuring stub routers

About this task

A stub router is used for traffic control. It reports its status as a stub router to neighboring OSPF routers. The neighboring routers can have a route to the stub router, but they do not use the stub router to forward data.

Router LSAs from the stub router might contain different link type values. A value of 3 means a link to a stub network, and the cost of the link will not be changed by default. To set the cost of the link to 65535, specify the **include-stub** keyword in the **stub-router** command. A value of 1, 2 or 4 means a point-to-point link, a link to a transit network, or a virtual link. On such links, a maximum cost value of 65535 is used. Neighbors do not send packets to the stub router as long as they have a route with a smaller cost.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Configure the router as a stub router.

   stub-router [**external-lsa** [*max-metric-value*] | **include-stub** | **on-startup** { *seconds* | **wait-for-bgp** [*seconds*] } | **summary-lsa** [*max-metric-value*]] *

   By default, the router is not configured as a stub router.

   A stub router is not related to a stub area.

Enabling compatibility with RFC 1583

About this task

RFC 1583 specifies a different method than RFC 2328 for selecting the optimal route to a destination in another AS. When multiple routes are available to the ASBR, OSPF selects the optimal route by using the following procedure:

1. Selects the route with the highest preference.
   - If RFC 2328 is compatible with RFC 1583, all these routes have equal preference.
   - If RFC 2328 is not compatible with RFC 1583, the intra-area route in a non-backbone area is preferred to reduce the burden of the backbone area. The inter-area route and intra-area route in the backbone area have equal preference.
2. Selects the route with the lower cost if two routes have equal preference.
3. Selects the route with the larger originating area ID if two routes have equal cost.

Restrictions and guidelines

To avoid routing loops, set identical RFC 1583-compatibility on all routers in a routing domain.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable compatibility with RFC 1583.

   rfc1583 compatible

   By default, compatibility with RFC 1583 is enabled.

Enabling OSPF to group ECMP routes

About this task

Perform this task to enable OSPF to group ECMP routes by prefix to speed up route convergence.

This feature is applicable to a network when the network has a large number of ECMP routes and different route prefixes in the network have the same next hops. For example, OSPF learns 10000 route prefixes and all route prefixes have the same 16 next hops (1.1.1.1 to 1.1.1.16). Without this feature, OSPF has to send all ECMP routes of every route prefix (10000 × 16 routes) to the route management module. After you enable this feature, OSPF groups the ECMP routes by prefix and sends the route groups (10000 route groups) to the route management module.

Restrictions and guidelines

If the output interfaces to the next hops of ECMP routes are TE tunnel interfaces, OSPF groups the ECMP routes regardless of whether you enable this feature or not.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable OSPF to group ECMP routes.

ecmp-group enable

By default, OSPF does not group ECMP routes.

Configuring OSPF GR

About OSPF GR

GR ensures forwarding continuity when a routing protocol restarts or an active/standby switchover occurs.

Two routers are required to complete a GR process. The following are router roles in a GR process:

- **GR restarter**—Graceful restarting router. It must have GR capability.
- **GR helper**—A neighbor of the GR restarter. It helps the GR restarter to complete the GR process.

OSPF GR has the following types:

- **IETF GR**—Uses Opaque LSAs to implement GR.
- **Non-IETF GR**—Uses link local signaling (LLS) to advertise GR capability and uses out of band synchronization to synchronize the LSDB.

A device can act as a GR restarter and GR helper at the same time.

Restrictions and guidelines for OSPF GR

You cannot enable OSPF NSR on a device that acts as GR restarter.

Configuring OSPF GR restarter

Configuring the IETF OSPF GR restarter

1. Enter system view.

   system-view

2. Enable OSPF and enter its view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable opaque LSA reception and advertisement capability.

   opaque-capability enable

   By default, opaque LSA reception and advertisement capability is enabled.

4. Enable the IETF GR.

   graceful-restart ietf [**global** | **planned-only**] *

   By default, the IETF GR capability is disabled.

5. (Optional.) Set the GR interval.

   graceful-restart interval *interval*

   By default, the GR interval is 120 seconds.

Configuring the non-IETF OSPF GR restarter

1. Enter system view.

   system-view

2. Enable OSPF and enter its view.

```
ospf [ process-id | router-id { auto-select | router-id } |
vpn-instance vpn-instance-name ] *
```

3. Enable the link-local signaling capability.

   ```
   enable link-local-signaling
   ```

   By default, the link-local signaling capability is disabled.

4. Enable the out-of-band re-synchronization capability.

   ```
   enable out-of-band-resynchronization
   ```

   By default, the out-of-band re-synchronization capability is disabled.

5. Enable non-IETF GR.

   ```
   graceful-restart [ nonstandard ] [ global | planned-only ] *
   ```

   By default, non-IETF GR capability is disabled.

6. (Optional.) Set the GR interval.

   ```
   graceful-restart interval interval
   ```

   By default, the GR interval is 120 seconds.

Configuring OSPF GR helper

Configuring the IETF OSPF GR helper

1. Enter system view.

   ```
   system-view
   ```

2. Enable OSPF and enter its view.

   ```
   ospf [ process-id | router-id { auto-select | router-id } |
   vpn-instance vpn-instance-name ] *
   ```

3. Enable opaque LSA reception and advertisement capability.

   ```
   opaque-capability enable
   ```

   By default, opaque LSA reception and advertisement capability is enabled.

4. Enable GR helper capability.

   ```
   graceful-restart helper enable [ planned-only ]
   ```

   By default, GR helper capability is enabled.

5. (Optional.) Enable strict LSA checking for the GR helper.

   ```
   graceful-restart helper strict-lsa-checking
   ```

   By default, strict LSA checking for the GR helper is disabled.

   When an LSA change on the GR helper is detected, the GR helper device exits the GR helper mode.

Configuring the non-IETF OSPF GR helper

1. Enter system view.

   ```
   system-view
   ```

2. Enable OSPF and enter its view.

   ```
   ospf [ process-id | router-id { auto-select | router-id } |
   vpn-instance vpn-instance-name ] *
   ```

3. Enable the link-local signaling capability.

   ```
   enable link-local-signaling
   ```

   By default, the link-local signaling capability is disabled.

4. Enable the out-of-band re-synchronization capability.

enable out-of-band-resynchronization

By default, the out-of-band re-synchronization capability is disabled.

5. Enable GR helper.

   graceful-restart helper enable

   By default, GR helper is enabled.

6. (Optional.) Enable strict LSA checking for the GR helper.

   graceful-restart helper strict-lsa-checking

   By default, strict LSA checking for the GR helper is disabled.

   When an LSA change on the GR helper is detected, the GR helper device exits the GR helper mode.

Triggering OSPF GR

About this task

You can trigger OSPF GR by performing an active/standby switchover or using the **reset ospf process** command.

Procedure

To trigger OSPF GR, execute the **reset ospf** [*process-id*] **process graceful-restart** command in user view.

Configuring OSPF NSR

About this task

Nonstop routing (NSR) backs up OSPF link state information from the active process to the standby process. After an active/standby switchover, NSR can complete link state recovery and route regeneration without tearing down adjacencies or impacting forwarding services.

NSR does not require the cooperation of neighboring devices to recover routing information, and it is typically used more often than GR.

Restrictions and guidelines

A device that has OSPF NSR enabled cannot act as GR restarter.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable OSPF NSR.

   non-stop-routing

   By default, OSPF NSR is disabled.

   This command takes effect only for the current process. As a best practice, enable OSPF NSR for each process if multiple OSPF processes exist.

Configuring BFD for OSPF

About BFD for OSPF

BFD provides a single mechanism to quickly detect and monitor the connectivity of links between OSPF neighbors, which improves the network convergence speed.For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

OSPF supports the following BFD detection modes:

- **Bidirectional control detection**—Requires BFD configuration to be made on both OSPF routers on the link.
- **Single-hop echo detection**—Requires BFD configuration to be made on one OSPF router on the link.

Configuring bidirectional control detection

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Enable BFD bidirectional control detection.
   ospf bfd enable

   By default, BFD bidirectional control detection is disabled.

   Both ends of a BFD session must be on the same network segment and in the same area.

Configuring single-hop echo detection

1. Enter system view.
   system-view
2. Configure the source address of echo packets.
   bfd echo-source-ip *ip-address*

   By default, the source address of echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interfaces.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.
3. Enter interface view.
   interface *interface-type interface-number*
4. Enable BFD single-hop echo detection.
   ospf bfd enable echo

   By default, BFD single-hop echo detection is disabled.

Configuring OSPF FRR

About OSPF FRR

OSPF Fast Reroute (FRR) calculates a backup path based on the LSDB or specifies a backup path by using a routing policy. Then, it saves the backup path information to the FIB. When the primary path fails, the system immediately switches traffic to the backup path to prevent traffic loss and reduce the route convergence time.

OSPF supports only Loop Free Alternate (LFA) FRR.

The following OSPF FRR traffic protection types are available:

- **Link protection**—Protects traffic that traverses a specific link.
- **Node protection**—Protects traffic that traverses a specific node.

Node protection takes precedence over link protection.

Restrictions and guidelines for OSPF FRR

If both OSPF FRR and PIC are configured, OSPF FRR takes effect.

OSPF FRR tasks at a glance

To configure OSPF FRR, perform the following tasks:

1. Configuring an OSPF backup path

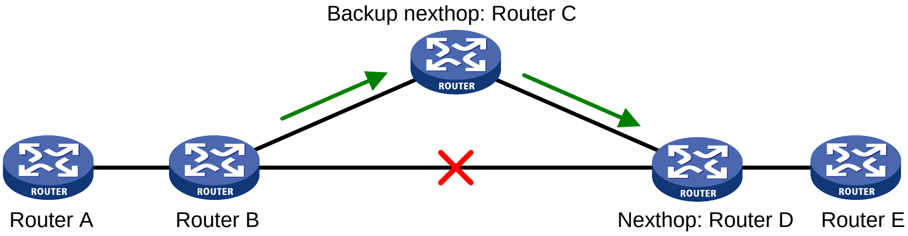
   Perform a minimum of one task.
   - Configuring OSPF LFA FRR
   - Configuring OSPF FRR to use a backup next hop specified in a routing policy
2. (Optional.) Setting the priority for FRR backup path selection policies
3. (Optional.) Configuring BFD for OSPF FRR
   - Configuring BFD control packet mode for OSPF FRR
   - Configuring BFD echo packet mode for OSPF FRR

Configuring OSPF LFA FRR

About this task

A link or router failure on a path can cause packet loss until OSPF completes routing convergence based on the new network topology. FRR enables fast rerouting to minimize the impact of link or node failures.

Figure 9 Network diagram for OSPF FRR



As shown in Figure 9, configure FRR on Router B by using a routing policy to specify a backup next hop. When the primary link fails, OSPF directs packets to the backup next hop. At the same time, OSPF calculates the shortest path based on the new network topology. It forwards packets over the path after network convergence.

You can configure OSPF FRR to calculate a backup next hop by using the loop free alternate (LFA) algorithm, or specify a backup next hop by using a routing policy.

Restrictions and guidelines for OSPF FRR

Do not use the **fast-reroute lfa** command together with the **vlink-peer** command.

Procedure

1. Enter system view.

   system-view
2. Enter interface view.

   interface *interface-type interface-number*
3. (Optional.) Enable LFA on an interface.

   ospf fast-reroute lfa-backup

   By default, the interface is enabled with LFA and it can be selected as a backup interface.
4. Return to system view.

   quit
5. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *
6. Enable OSPF FRR to use the LFA algorithm to calculate a backup next hop.

   fast-reroute lfa [**abr-only**]

   By default, OSPF FRR is disabled.

   If **abr-only** is specified, only the route to the ABR is selected as the backup path.

Configuring OSPF FRR to use a backup next hop specified in a routing policy

About this task

Before you perform this task, use the **apply fast-reroute backup-interface** command to specify a backup next hop in a routing policy for OSPF FRR. For more information about the **apply fast-reroute backup-interface** command and routing policy configuration, see "Configuring routing policies."

Procedure

1. Enter system view.

   system-view
2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *
3. Enable OSPF FRR to use a backup next hop specified in a routing policy.

   fast-reroute route-policy *route-policy-name*

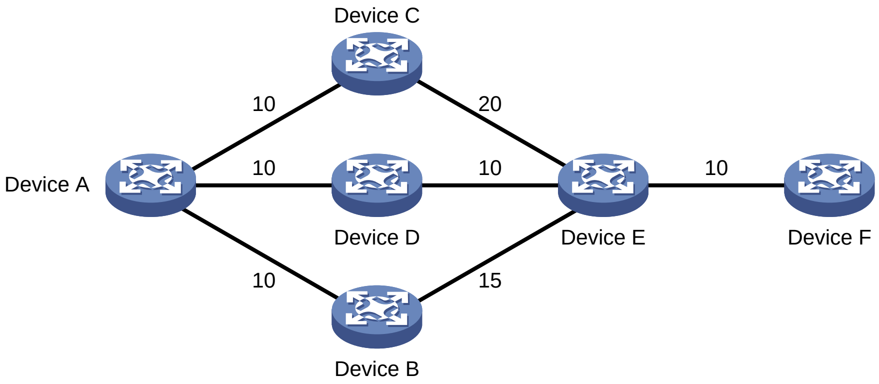
   By default, OSPF FRR is disabled.

Setting the priority for FRR backup path selection policies

About this task

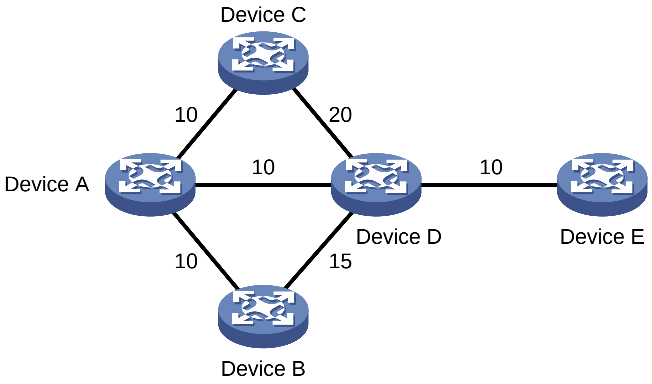
By default, OSPF uses the node-protection policy to select the backup path. As shown in Figure 10, traffic forwarding path **Device A->Device D->Device E->Device F** has two backup paths: **Device A->Device C->Device E->Device F** and **Device A->Device B->Device E->Device F**. Because Device C has higher forwarding capability than Device B, OSPF prefers backup path **Device A->Device C->Device E->Device F**.

Figure 10 Node-protection backup path selection



To apply the lowest-cost policy that is more applicable in certain networks, perform this task to set a higher priority for it than the node-protection policy. As shown in Figure 11, traffic forwarding path **Device A->Device D->Device E** has two backup paths: **Device A->Device C->Device D->Device E** and **Device A->Device B->Device D->Device E**. By default, OSPF selects backup path **Device A->Device C->Device D->Device E** according to the node-protection policy. For OSPF to select backup path **Device A->Device B->Device D->Device E**, set a higher priority for the lowest-cost policy than the node-protection policy.

Figure 11 Lowest-cost backup path selection



Restrictions and guidelines

If the node-protection policy has the higher priority but the backup path calculation still fails, OSPF uses the lowest-cost policy for further calculation.

If the lowest-cost policy has the higher priority but the backup path calculation still fails, OSPF does not perform further backup path calculation.

Procedure

1. Enter system view.

system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the priority for the node-protection or lowest-cost backup path selection policy.

   fast-reroute tiebreaker { **lowest-cost** | **node-protecting** } **preference** *preference*

   By default, the priority values of the node-protection and lowest-cost backup path selection policies are 40 and 20, respectively.

Configuring BFD control packet mode for OSPF FRR

About this task

By default, OSPF FRR does not use BFD to detect primary link failures. To speed up OSPF convergence, enable BFD control packet mode for OSPF FRR to detect primary link failures. This mode requires BFD configuration on both OSPF routers on the link.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for OSPF FRR.

   ospf primary-path-detect bfd ctrl

   By default, BFD control packet mode is disabled for OSPF FRR.

Configuring BFD echo packet mode for OSPF FRR

About this task

By default, OSPF FRR does not use BFD to detect primary link failures. To speed up OSPF convergence, enable BFD echo packet mode for OSPF FRR to detect primary link failures. This mode requires BFD configuration on one OSPF router on the link.

Procedure

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interfaces.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD echo packet mode for OSPF FRR.

   ospf primary-path-detect bfd echo

   By default, BFD echo packet mode is disabled for OSPF FRR.

Configuring OSPF authentication

About OSPF area and interface authentication

Perform this task to configure OSPF area and interface authentication.

OSPF adds the configured key into sent packets, and uses the key to authenticate received packets. Only packets that pass the authentication can be received. If a packet fails the authentication, the OSPF neighbor relationship cannot be established.

If you configure OSPF authentication for both an area and an interface in that area, the interface uses the OSPF authentication configured on it.

Configuring OSPF area authentication

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enter area view.

   area *area-id*

4. Configure area authentication mode.
   - Configure HMAC-MD5/MD5 authentication.

     authentication-mode { **hmac-md5** | **hmac-sha-256** | **md5** } *key-id* { **cipher** | **plain** } *string*
   - Configure simple authentication.

     authentication-mode simple { **cipher** | **plain** } *string*
   - Configure keychain authentication.

     authentication-mode keychain *keychain-name*

     For more information about keychains, see keychain configuration in *Security Configuration Guide*.

   By default, no authentication is configured.

   You must configure the same authentication mode and key on all the routers in an area.

Configuring OSPF interface authentication

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure interface authentication mode.
   - Configure simple authentication.

     ospf authentication-mode simple { **cipher** | **plain** } *string*
   - Configure HMAC-MD5/MD5 authentication.

     ospf authentication-mode { **hmac-md5** | **hmac-sha-256** | **md5** } *key-id* { **cipher** | **plain** } *string*

- Configure keychain authentication.

  ospf authentication-mode keychain *keychain-name*

  For information about keychains, see *Security Configuration Guide*.

By default, no authentication is configured.

You must configure the same authentication mode and key on both the local interface and its peer interface.

Configuring GTSM for OSPF

About GTSM

The Generalized TTL Security Mechanism (GTSM) protects the device by comparing the TTL value in the IP header of incoming OSPF packets against a valid TTL range. If the TTL value is within the valid TTL range, the packet is accepted. If not, the packet is discarded.

The valid TTL range is from 255 – the configured hop count + 1 to 255.

When GTSM is configured, the OSPF packets sent by the device have a TTL of 255.

Restrictions and guidelines for GTSM

To use GTSM, you must configure GTSM on both the local and peer devices. You can specify different *hop-count* values for them.

The configuration in OSPF area view applies to all OSPF interfaces in the area. The configuration in interface view takes precedence over the configuration in OSPF area view.

Configuring GTSM in OSPF area view

1. Enter system view.

   system-view
2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *
3. Enter OSPF area view.

   area *area-id*
4. Enable GTSM for the OSPF area.

   ttl-security [**hops** *hop-count*]

   By default, GTSM is disabled for the OSPF area.

Configuring GTSM in interface view

1. Enter system view.

   system-view
2. Enter interface view.

   interface *interface-type interface-number*
3. Configure GTSM for the interface.
   - Enable GTSM for the interface.

     ospf ttl-security [**hops** *hop-count*]

- Disable GTSM for the interface.

  ospf ttl-security disable

  Disable GTSM for an interface when the following conditions exist:

  - The area to which the interface belongs is enabled with GTSM.
  - The neighbor of the interface does not support GTSM.

By default, an interface uses the GTSM configuration of the area to which the interface belongs.

Configuring OSPF logging and SNMP notifications

Logging neighbor state changes

About this task

Perform this task to enable output of neighbor state change logs to the information center. The information center processes the logs according to user-defined output rules (whether and where to output logs). For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Enable the logging of neighbor state changes.

   log-peer-change

   By default, this feature of logging neighbor state changes is enabled.

Configuring the OSPF logging feature

About this task

OSPF logs include LSA aging logs, route calculation logs, neighbor logs, OSPF route logs, and self-originated and received LSA logs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

3. Set the number of OSPF logs.

   event-log { **hello** { **received** [**abnormal** | **dropped**] | **sent** [**abnormal** | **failed**] } | **lsa-flush** | **peer** | **spf** } **size** *count*

   By default, the device can generate a maximum of 100 OSPF logs for each type.

Configuring OSPF network management

About this task

This task involves the following configurations:

- Bind an OSPF process to MIB so that you can use network management software to manage the specified OSPF process.
- Enable SNMP notifications for OSPF to report important events.
- Configure the SNMP notification output interval and the maximum number of SNMP notifications that can be output at each interval.

To report critical OSPF events to an NMS, enable SNMP notifications for OSPF. For SNMP notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see the network management and monitoring configuration guide for the device.

Procedure

1. Enter system view.

   system-view

2. Bind MIB to an OSPF process.

   ospf mib-binding *process-id*

   By default, MIB is bound to the process with the smallest process ID.

3. Enable SNMP notifications for OSPF.

   snmp-agent trap enable ospf [**authentication-failure** | **bad-packet** | **config-error** | **grhelper-status-change** | **grrestarter-status-change** | **if-state-change** | **lsa-maxage** | **lsa-originate** | **lsdb-approaching-overflow** | **lsdb-overflow** | **neighbor-state-change** | **nssatranslator-status-change** | **retransmit** | **virt-authentication-failure** | **virt-bad-packet** | **virt-config-error** | **virt-retransmit** | **virtgrhelper-status-change** | **virtif-state-change** | **virtneighbor-state-change**] *

   By default, SNMP notifications for OSPF are enabled.

4. Enter OSPF view.

   ospf [*process-id* | **router-id** { **auto-select** | *router-id* } | **vpn-instance** *vpn-instance-name*] *

5. Configure the SNMP notification output interval and the maximum number of SNMP notifications that can be output at each interval.

   snmp trap rate-limit interval *trap-interval* **count** *trap-number*

   By default, OSPF outputs a maximum of seven SNMP notifications within 10 seconds.

Display and maintenance commands for OSPF

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display summary route information on the OSPF ABR.	**display ospf** [*process-id*] [**area** *area-id*] **abr-summary** [*ip-address* { *mask-length* | *mask* }] [**verbose**]
Display OSPF FRR backup next hop information.	**display ospf** [*process-id*] [**area** *area-id*] **fast-reroute lfa-candidate**

Task	Command
Display OSPF SPF information.	**display ospf** [*process-id*] [**area** *area-id*] **spf-tree** [**verbose**]
Display OSPF process information.	**display ospf** [*process-id*] [**verbose**]
Display OSPF ABR and ASBR information.	**display ospf** [*process-id*] **abr-asbr** [**verbose**]
Display OSPF ASBR route summarization information.	**display ospf** [*process-id*] **asbr-summary** [*ip-address* { *mask-length* | *mask* }]
Display OSPF log information.	**display ospf** [*process-id*] **event-log** { **lsa-flush** | **peer** [*neighbor-id*] [**slot** *slot-number*] | **spf** }
Display OSPF log information about received or sent hello packets.	display ospf [process-id] event-log **hello** { **received** [**abnormal** | **dropped**] | **sent** } [*neighbor-id*] [**slot** *slot-number*] **display ospf** [*process-id*] **event-log hello sent** { **abnormal** | **failed** } [*neighbor-address*] [**slot** *slot-number*]
Display OSPF GR information.	**display ospf** [*process-id*] **graceful-restart** [**verbose**]
Display the router ID-to-host name mapping table.	**display ospf** [*process-id*] **hostname-table**
Display OSPF interface information.	**display ospf** [*process-id*] **interface** [*interface-type interface-number* | **verbose**]
Display information about hello packets sent by OSPF interfaces.	**display ospf** [*process-id*] **interface** [*interface-type interface-number*] **hello**
Display OSPF LSDB information.	**display ospf** [*process-id*] [**area** *area-id*] **lsdb** { **asbr** | **network** | **nssa** | **opaque-link** | **router** | **summary** } [*link-state-id*] [**originate-router** *advertising-router-id* | **self-originate**] [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *] [**resolve-hostname**] **display ospf** [*process-id*] [**area** *area-id*] **lsdb** { **asbr** | **network** | **nssa** | **opaque-link** | **router** | **summary** } [*link-state-id*] **hostname** *host-name* [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *] **display ospf** [*process-id*] **lsdb** [**brief** | **originate-router** *advertising-router-id* | **self-originate**] [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *] [**resolve-hostname**] **display ospf** [*process-id*] **lsdb hostname** *host-name* [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *] **display ospf** [*process-id*] **lsdb** { **ase** | **opaque-as** } [*link-state-id*] [**originate-router** *advertising-router-id* |

Task	Command
	self-originate] [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *] [**resolve-hostname**] **display ospf** [*process-id*] **lsdb** { **ase** | **opaque-as** } [*link-state-id*] **hostname** *host-name* [**age** { **max-value** *max-age-value* | **min-value** *min-age-value* } *]
Display OSPF next hop information.	**display ospf** [*process-id*] **nexthop**
Display OSPF NSR information.	**display ospf** [*process-id*] **non-stop-routing status**
Display OSPF neighbor information.	**display ospf** [*process-id*] **peer** [**hello** | **verbose**] [*interface-type interface-number*] [[*neighbor-id*] [**resolve-hostname**] | **hostname** *host-name*]
Display neighbor statistics for OSPF areas.	**display ospf** [*process-id*] **peer statistics**
Display OSPF request queue information.	**display ospf** [*process-id*] **request-queue** [*interface-type interface-number*] [*neighbor-id*]
Display OSPF retransmission queue information.	**display ospf** [*process-id*] **retrans-queue** [*interface-type interface-number*] [*neighbor-id*]
Display OSPF routing table information.	**display ospf** [*process-id*] **routing** [*ip-address* { *mask-length* | *mask* }] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*] [**verbose**]
Display OSPF statistics.	**display ospf** [*process-id*] **statistics** [**error** | **packet** [**hello** | *interface-type interface-number*]]
Display OSPF virtual link information.	**display ospf** [*process-id*] **vlink**
Display the global route ID.	**display router id**
Clear OSPF log information.	**reset ospf** [*process-id*] **event-log** [**lsa-flush** | **peer** [**slot** slot-number] | **spf**]
Clear OSPF log information about received or sent hello packets.	**reset ospf** [*process-id*] **event-log hello** { **received** [**abnormal** | **dropped**] | **sent** [**abnormal** | **failed**] } [**slot** *slot-number*]
Restart an OSPF process.	**reset ospf** [*process-id*] **process** [**graceful-restart**]
Re-enable OSPF route redistribution.	**reset ospf** [*process-id*] **redistribution**
Clear OSPF statistics.	**reset ospf** [*process-id*] **statistics**

OSPF configuration examples

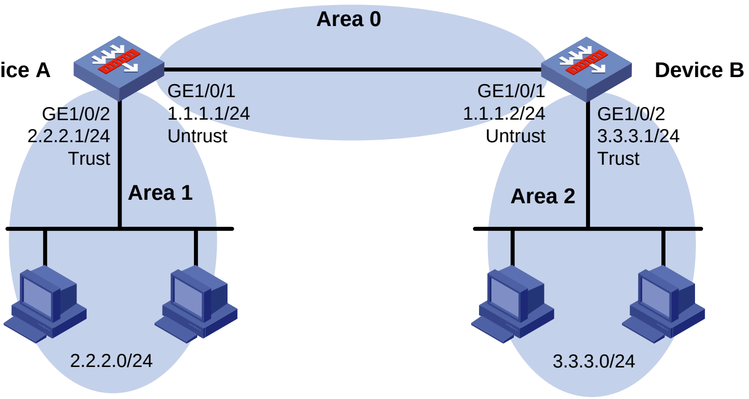
Example: Configuring basic OSPF

Network configuration

As shown in Figure 12:

- Enable OSPF on all devices, and split the AS into three areas.
- Configure Device A and Device B as ABRs.

Figure 12 Network diagram



Procedure

# Configure Device A.

1. Configure IP addresses for interfaces correctly according to Figure 12.
   a. Configure an IP address for GigabitEthernet 1/0/1.
```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0
[DeviceA-GigabitEthernet1/0/1] quit
```
   b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)
2. Add each interface on Device A to a security zone.
```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Trust] quit
```
3. Configure security policies.
   a. Configure a security policy to allow OSPF neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **ospflocalin** and permit Device A to receive OSPF packets from Device B.
```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name ospflocalin
[DeviceA-security-policy-ip-0-ospflocalin] source-zone untrust
```

```
[DeviceA-security-policy-ip-0-ospflocalin] destination-zone local
[DeviceA-security-policy-ip-0-ospflocalin] service ospf
[DeviceA-security-policy-ip-0-ospflocalin] action pass
[DeviceA-security-policy-ip-0-ospflocalin] quit
```

# Create security policy rule **ospflocalout** and permit Device A to send OSPF packets to Device B.

```
[DeviceA-security-policy-ip] rule name ospflocalout
[DeviceA-security-policy-ip-1-ospflocalout] source-zone local
[DeviceA-security-policy-ip-1-ospflocalout] destination-zone untrust
[DeviceA-security-policy-ip-1-ospflocalout] service ospf
[DeviceA-security-policy-ip-1-ospflocalout] action pass
[DeviceA-security-policy-ip-1-ospflocalout] quit
```

b. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust**.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-2-trust-untrust] source-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-2-trust-untrust] destination-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-2-trust-untrust] action pass
[DeviceA-security-policy-ip-2-trust-untrust] quit
```

# Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-3-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-3-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-3-untrust-trust] source-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-3-untrust-trust] destination-ip-subnet 2.2.2.0 24
[DeviceA-security-policy-ip-3-untrust-trust] action pass
[DeviceA-security-policy-ip-3-untrust-trust] quit
[DeviceA-security-policy-ip] quit
```

4. Enable basic OSPF functions.

```
[DeviceA] router id 2.2.2.1
[DeviceA] ospf
[DeviceA-ospf-1] area 0
[DeviceA-ospf-1-area-0.0.0.0] network 1.1.1.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.0] quit
[DeviceA-ospf-1] area 1
[DeviceA-ospf-1-area-0.0.0.1] network 2.2.2.0 0.0.0.255
[DeviceA-ospf-1-area-0.0.0.1] quit
[DeviceA-ospf-1] quit
```

# Configure Device B.

1. Configure IP addresses for interfaces correctly according to Figure 12.

a. Configure an IP address for GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
```

```
[DeviceB-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

   b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device B to a security zone.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.

   a. Configure a security policy to allow OSPF neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **ospflocalin** and permit Device B to receive OSPF packets from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ospflocalin
[DeviceB-security-policy-ip-0-ospflocalin] source-zone untrust
[DeviceB-security-policy-ip-0-ospflocalin] destination-zone local
[DeviceB-security-policy-ip-0-ospflocalin] service ospf
[DeviceB-security-policy-ip-0-ospflocalin] action pass
[DeviceB-security-policy-ip-0-ospflocalin] quit
```

   # Create security policy rule **ospflocalout** and permit Device B to send OSPF packets to Device A.

```
[DeviceB-security-policy-ip] rule name ospflocalout
[DeviceB-security-policy-ip-1-ospflocalout] source-zone local
[DeviceB-security-policy-ip-1-ospflocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ospflocalout] service ospf
[DeviceB-security-policy-ip-1-ospflocalout] action pass
[DeviceB-security-policy-ip-1-ospflocalout] quit
```

   b. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust**.

   # Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-2-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-2-trust-untrust] source-ip-subnet 3.3.3.0 24
[DeviceB-security-policy-ip-2-trust-untrust] destination-ip-subnet 2.2.2.0 24
[DeviceB-security-policy-ip-2-trust-untrust] action pass
[DeviceB-security-policy-ip-2-trust-untrust] quit
```

   # Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-3-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-3-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-3-untrust-trust] source-ip-subnet 2.2.2.0 24
[DeviceB-security-policy-ip-3-untrust-trust] destination-ip-subnet 3.3.3.0 24
```

```
   [DeviceB-security-policy-ip-3-untrust-trust] action pass
   [DeviceB-security-policy-ip-3-untrust-trust] quit
   [DeviceB-security-policy-ip] quit
```

4. Enable basic OSPF functions.

```
[DeviceB] router id 3.3.3.1
[DeviceB] ospf
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 1.1.1.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] area 2
[DeviceB-ospf-1-area-0.0.0.2] network 3.3.3.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.2] quit
[DeviceB-ospf-1] quit
```

Verifying the configuration

# View detailed information about OSPF neighbors on Device A.

```
[DeviceA] display ospf peer verbose

         OSPF Process 1 with Router ID 2.2.2.1
                 Neighbors

 Area 0.0.0.0 interface 1.1.1.1(GigabitEthernet1/0/1)'s neighbors
 Router ID: 3.3.3.1          Address: 1.1.1.2          GR State: Normal
   State: Full  Mode: Nbr is master  Priority: 1
   DR: 1.1.1.1  BDR: 1.1.1.2  MTU: 0
   Options is 0x42 (-|O|-|-|-|-|E|-)
   Dead timer due in 32  sec
   Neighbor is up for 00:07:08
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
   BFD status: Disabled
```

# View OSPF routing information on Device A.

```
[DeviceA] display ospf routing

         OSPF Process 1 with Router ID 2.2.2.1
                  Routing Table

 Routing for network
 Destination        Cost     Type    NextHop         AdvRouter       Area
 3.3.3.0/24         2        Inter   1.1.1.2         3.3.3.1         0.0.0.0
 2.2.2.0/24         1        Stub    0.0.0.0         2.2.2.1         0.0.0.1
 1.1.1.0/24         1        Transit 0.0.0.0         2.2.2.1         0.0.0.0

 Total nets: 3
 Intra area: 2  Inter area: 1  ASE: 0  NSSA: 0
```

# Verify that hosts in area 1 can ping hosts in area 2, and vice versa.

Contents

Configuring OSPFv3

About OSPFv3

This chapter describes how to configure RFC 2740-compliant Open Shortest Path First version 3 (OSPFv3) for an IPv6 network.

Comparison of OSPFv3 with OSPFv2

OSPFv3 and OSPFv2 have the following in common:

- 32-bit router ID and area ID.
- Hello, Database Description (DD), Link State Request (LSR), Link State Update (LSU), Link State Acknowledgment (LSAck).
- Mechanisms for finding neighbors and establishing adjacencies.
- Mechanisms for advertising and aging LSAs.

OSPFv3 and OSPFv2 have the following differences:

- OSPFv3 runs on a per-link basis. OSPFv2 runs on a per-IP-subnet basis.
- OSPFv3 supports running multiple processes on an interface, but OSPFv2 does not support.
- OSPFv3 identifies neighbors by router ID. OSPFv2 identifies neighbors by IP address.

For more information about OSPFv2, see "Configuring OSPF."

OSPFv3 packets

OSPFv3 uses the following packet types:

- **Hello**—Periodically sent to find and maintain neighbors, containing timer values, information about the DR, BDR, and known neighbors.
- **DD**—Describes the digest of each LSA in the LSDB, exchanged between two routers for data synchronization.
- **LSR**—Requests needed LSAs from the neighbor. After exchanging the DD packets, the two routers know which LSAs of the neighbor are missing from their LSDBs. They then send an LSR packet to each other, requesting the missing LSAs. The LSA packet contains the digest of the missing LSAs.
- **LSU**—Transmits the requested LSAs to the neighbor.
- **LSAck**—Acknowledges received LSU packets.

OSPFv3 LSA types

OSPFv3 sends routing information in LSAs. The following LSAs are commonly used:

- **Router LSA**—Type-1 LSA, originated by all routers. This LSA describes the collected states of the router's interfaces to an area, and is flooded throughout a single area only.
- **Network LSA**—Type-2 LSA, originated for broadcast and NBMA networks by the DR. This LSA contains the list of routers connected to the network, and is flooded throughout a single area only.
- **Inter-Area-Prefix LSA**—Type-3 LSA, originated by ABRs and flooded throughout the LSA's associated area. Each Inter-Area-Prefix LSA describes a route with IPv6 address prefix to a destination outside the area, yet still inside the AS.

- **Inter-Area-Router LSA**—Type-4 LSA, originated by ABRs and flooded throughout the LSA's associated area. Each Inter-Area-Router LSA describes a route to ASBR.
- **AS External LSA**—Type-5 LSA, originated by ASBRs, and flooded throughout the AS, except stub areas and Not-So-Stubby Areas (NSSAs). Each AS External LSA describes a route to another AS. A default route can be described by an AS External LSA.
- **NSSA LSA**—Type-7 LSA, originated by ASBRs in NSSAs and flooded throughout a single NSSA. NSSA LSAs describe routes to other ASs.
- **Link LSA**—Type-8 LSA. A router originates a separate Link LSA for each attached link. Link LSAs have link-local flooding scope. Each Link LSA describes the IPv6 address prefix of the link and Link-local address of the router.
- **Intra-Area-Prefix LSA**—Type-9 LSA. Each Intra-Area-Prefix LSA contains IPv6 prefix information on a router, stub area, or transit area information, and has area flooding scope. It was introduced because Router LSAs and Network LSAs contain no address information.
- **Grace LSA**—Type-11 LSA, generated by a GR restarter at reboot and transmitted on the local link. The GR restarter describes the cause and interval of the reboot in the Grace LSA to notify its neighbors that it performs a GR operation.

Protocols and standards

- RFC 2328, *OSPF Version 2*
- RFC 3101, *OSPF Not-So-Stubby Area (NSSA) Option*
- RFC 4552, *Authentication/Confidentiality for OSPFv3*
- RFC 5187, *OSPFv3 Graceful Restart*
- RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*
- RFC 5329, *Traffic Engineering Extensions to OSPF Version 3*
- RFC 5340, *OSPF for IPv6*
- RFC 5523, *OSPFv3-Based Layer 1 VPN Auto-Discovery*
- RFC 5643, *Management Information Base for OSPFv3*
- RFC 6506, *Supporting Authentication Trailer for OSPFv3*
- RFC 6565, *OSPFv3 as a Provider Edge to Customer Edge (PE-CE) Routing Protocol*
- RFC 6969, *OSPFv3 Instance ID Registry Update*
- RFC 7166, *Supporting Authentication Trailer for OSPFv3*

OSPFv3 tasks at a glance

To configure OSPFv3, perform the following tasks:

1. Enabling OSPFv3
2. (Optional.) Configuring OSPFv3 area parameters
   - Configuring a stub area
   - Configuring an NSSA area
   - Configuring an OSPFv3 virtual link

     Perform this task on an ABR to create a virtual link when connectivity cannot be maintained between a non-backbone area and the backbone, or within the backbone.
3. (Optional.) Configuring OSPFv3 network types
   - Setting the broadcast network type for an OSPFv3 interface
   - Setting the NBMA network type for an OSPFv3 interface

   - Setting the P2MP network type for an OSPFv3 interface
   - Setting the P2P network type for an OSPFv3 interface
4. (Optional.) Configuring OSPFv3 route control
   - Configuring OSPFv3 inter-area route summarization
   - Configuring OSPFv3 received route filtering
   - Configuring Inter-Area-Prefix LSA filtering
   - Setting an OSPFv3 cost for an interface
   - Setting the maximum number of OSPFv3 ECMP routes
   - Setting a preference for OSPFv3
   - Configuring OSPFv3 route redistribution
5. (Optional.) Setting OSPFv3 timers
   - Setting OSPFv3 packet timers
   - Setting LSA transmission delay
   - Setting SPF calculation interval
   - Setting the LSA generation interval
   - Setting the LSU transmit rate
6. (Optional.) Setting a DR priority for an interface
7. (Optional.) Configuring OSPFv3 packet parameters
   - Ignoring MTU check for DD packets
   - Disabling interfaces from receiving and sending OSPFv3 packets
8. (Optional.) Configuring prefix suppression
9. (Optional.) Configuring a stub router
10. (Optional.) Enhancing OSPFv3 availability
    - Configuring OSPFv3 GR
    - Configuring OSPFv3 NSR
    - Configuring BFD for OSPFv3
    - Configuring OSPFv3 FRR
11. (Optional.) Configuring OSPFv3 logging and SNMP notifications
    - Enabling logging for neighbor state changes
    - Setting the maximum number of OSPFv3 logs
    - Configuring OSPFv3 network management

Enabling OSPFv3

About this task

To enable an OSPFv3 process on a router:

1. Enable the OSPFv3 process globally.
2. Assign the OSPFv3 process a router ID.
3. Enable the OSPFv3 process on related interfaces.

An OSPFv3 process ID has only local significance. Process 1 on a router can exchange packets with process 2 on another router.

OSPFv3 requires you to manually specify a router ID for each router in an AS. Make sure all assigned router IDs in the AS are unique.

Restrictions and guideline

If a router runs multiple OSPFv3 processes, specify a unique router ID for each process as a best practice.

Procedure

1. Enter system view.
   system-view
2. Enable an OSPFv3 process and enter its view.
   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *
   By default, no OSPFv3 processes are enabled.
3. Specify a router ID.
   router-id *router-id*
   By default, no router ID is configured.
4. Enter interface view.
   interface *interface-type interface-number*
5. Enable an OSPFv3 process on the interface.
   ospfv3 *process-id* **area** *area-id* [**instance** *instance-id*]
   By default, no OSPFv3 processes are enabled on an interface.

Configuring OSPFv3 area parameters

About OSPFv3 areas

OSPFv3 has the same stub area, NSSA area, and virtual link features as OSPFv2.

After you split an OSPFv3 AS into multiple areas, the LSA number is reduced and OSPFv3 applications are extended. To further reduce the size of routing tables and the number of LSAs, configure the non-backbone areas at an AS edge as stub areas.

A stub area cannot import external routes, but an NSSA area can import external routes into the OSPFv3 routing domain while retaining other stub area characteristics.

Non-backbone areas exchange routing information through the backbone area, so the backbone and non-backbone areas (including the backbone itself) must be fully meshed. If no connectivity can be achieved, configure virtual links.

Configuring a stub area

Restrictions and guidelines

To configure a stub area, you must perform this task on all routers attached to the area.

Procedure

1. Enter system view.
   system-view
2. Enter OSPFv3 view.
   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *
3. Enter OSPFv3 area view.
   area *area-id*
4. Configure the area as a stub area.

stub [**default-route-advertise-always** | **no-summary**] *

By default, no area is configured as a stub area.

The **no-summary** keyword is only available on the ABR of a stub area. If you specify the **no-summary** keyword, the ABR only advertises a default route in an Inter-Area-Prefix LSA into the stub area.

5. (Optional.) Set a cost for the default route advertised to the stub area.

   default-cost *cost-value*

   By default, the cost for the default route advertised to the stub area is 1.

Configuring an NSSA area

Restrictions and guidelines

To configure an NSSA area, you must perform this task on all routers attached to the area.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enter OSPFv3 area view.

   area *area-id*

4. Configure the area as an NSSA area.

   nssa [**default-route-advertise** [**cost** *cost-value* | **nssa-only** | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] * | **no-import-route** | **no-summary** | [**translate-always** | **translate-never**] | **suppress-fa** | **translator-stability-interval** *value*] *

   By default, no area is configured as an NSSA area.

   To configure a totally NSSA area, execute the **nssa no-summary** command on the ABR. The ABR of a totally NSSA area does not advertise inter-area routes into the area.

5. (Optional.) Set a cost for the default route advertised to the NSSA area.

   default-cost *cost-value*

   By default, the cost for the default route advertised to the NSSA area is 1.

   This command takes effect only on the ABR/ASBR of an NSSA or totally NSSA area.

Configuring an OSPFv3 virtual link

About this task

You can configure a virtual link to maintain connectivity between a non-backbone area and the backbone, or in the backbone itself.

Restrictions and guidelines

Both ends of a virtual link are ABRs that must be configured with the **vlink-peer** command.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enter OSPFv3 area view.

   area *area-id*

4. Configure a virtual link.

   vlink-peer *router-id* [**dead** *seconds* | **hello** *seconds* | **instance** *instance-id* | **ipsec-profile** *profile-name* | **keychain** *keychain-name* | **retransmit** *seconds* | **trans-delay** *seconds*] *

Configuring OSPFv3 network types

Restrictions and guidelines for OSPFv3 network type configuration

Based on the link layer protocol, OSPFv3 classifies networks into different types, including broadcast, NBMA, P2MP, and P2P.

- If any routers in a broadcast network do not support multicasting, you can change the network type to NBMA.
- An NBMA network must be fully connected. Any two routers in the network must be directly reachable to each other through a virtual circuit. If no such direct link is available, you must change the network type through a command.
- If direct connections are not available between some routers on an NBMA network, the type of interfaces associated must be configured as P2MP.
- If only two routers running OSPFv3 exist on a network segment, you can change the network type to P2P to save costs.

Setting the broadcast network type for an OSPFv3 interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the network type to broadcast for the OSPFv3 interface.

   ospfv3 network-type broadcast [**instance** *instance-id*]

   By default, the network type of an interface depends on the media type of the interface.

   When the link layer protocol is Ethernet or FDDI, the network type is broadcast by default.

Setting the NBMA network type for an OSPFv3 interface

Restrictions and guidelines

For NBMA interfaces, you must specify the link-local IP addresses and DR priorities for their neighbors because these interfaces cannot find neighbors by broadcasting hello packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the network type to NBMA for the OSPFv3 interface.

   ospfv3 network-type nbma [**instance** *instance-id*]

4. (Optional.) Set the router priority for the interface

   ospfv3 dr-priority *priority*

   By default, an interface has a router priority of 1.

   An interface's router priority determines its privilege in DR/BDR selection.

5. Specify an NBMA neighbor.

   ospfv3 peer *ipv6-address* [**cost** *cost-value* | **dr-priority** *priority*] [**instance** *instance-id*]

   By default, no link-local address is specified for the neighbor interface.

Setting the P2MP network type for an OSPFv3 interface

Restrictions and guidelines

For P2MP interfaces (only when in unicast mode), you must specify the link-local IP addresses of their neighbors because these interfaces cannot find neighbors by broadcasting hello packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the network type to P2MP for the OSPFv3 interface.

   ospfv3 network-type p2mp [**unicast**] [**instance** *instance-id*]

   By default, the network type of an interface depends on the media type of the interface.

4. Specify a P2MP unicast neighbor.

   ospfv3 peer *ipv6-address* [**cost** *cost-value* | **dr-priority** *priority*] [**instance** *instance-id*]

   By default, no link-local address is specified for the neighbor interface.

Setting the P2P network type for an OSPFv3 interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the network type to P2P for the OSPFv3 interface.

   ospfv3 network-type p2p [**instance** *instance-id*]

   When the link layer protocol is PPP, the network type is P2P by default.

Configuring OSPFv3 route control

Configuring OSPFv3 inter-area route summarization

About this task

If contiguous network segments exist in an area, you can summarize them into one network segment on the ABR. The ABR will advertise only the summary route. Any LSA on the specified network segment will not be advertised, reducing the LSDB size in other areas.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enter OSPFv3 area view.

   area *area-id*

4. Configure route summarization on the ABR.

   abr-summary *ipv6-address prefix-length* [**not-advertise**] [**cost** *cost-value*]

   By default, route summarization is not configured on an ABR.

Configuring redistributed route summarization

About this task

Perform this task to enable an ASBR to summarize external routes within the specified address range into a single route.

An ASBR can summarize routes in the following LSAs:

- Type-5 LSAs.
- Type-7 LSAs in an NSSA area.
- Type-5 LSAs translated from Type-7 LSAs in an NSSA area if the ASBR (also an ABR) is a translator. If the ASBR is not a translator, it cannot summarize routes in Type-5 LSAs translated from Type-7 LSAs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Configure route summarization on an ASBR.

   asbr-summary *ipv6-address prefix-length* [**cost** *cost-value* | **not-advertise** | **nssa-only** | **tag** *tag*] *

   By default, route summarization is not configured on an ASBR.

Configuring OSPFv3 received route filtering

About this task

This task allows you to filter routes calculated by using received LSAs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Configure OSPFv3 to filter routes calculated by using received LSAs.

   filter-policy { *ipv6-acl-number* [**gateway** *prefix-list-name*] | **prefix-list** *prefix-list-name* [**gateway** *prefix-list-name*] | **gateway** *prefix-list-name* | **route-policy** *route-policy-name* } **import**

   By default, OSPFv3 accepts all routes calculated by using received LSAs.

   This command can only filter routes computed by OSPFv3. Only routes not filtered out can be added into the local routing table.

Configuring Inter-Area-Prefix LSA filtering

Restrictions and guidelines

The **filter** command takes effect only on ABRs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enter OSPFv3 area view.

   area *area-id*

4. Configure OSPFv3 to filter Inter-Area-Prefix LSAs.

   filter { *ipv6-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } { **export** | **import** }

   By default, OSPFv3 accepts all Inter-Area-Prefix LSAs.

Setting an OSPFv3 cost for an interface

About this task

You can set an OSPFv3 cost for an interface with one of the following methods:

- Set the cost value in interface view.
- Set a bandwidth reference value for the interface, and OSPFv3 computes the cost automatically based on the bandwidth reference value by using the following formula:

  Interface OSPFv3 cost = Bandwidth reference value (100 Mbps) / Interface bandwidth (Mbps)

  - If the calculated cost is greater than 65535, the value of 65535 is used.
  - If the calculated cost is smaller than 1, the value of 1 is used.
- If no cost is set for an interface, OSPFv3 automatically computes the cost for the interface.

Setting a cost in interface view

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set an OSPFv3 cost for the interface.

   ospfv3 cost *cost-value* [**instance** *instance-id*]

   By default, the OSPFv3 cost is 1 for a VLAN interface and 0 for a loopback interface. The OSPFv3 cost is automatically computed according to the interface bandwidth for other interfaces.

Setting a bandwidth reference value

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set a bandwidth reference value.

   bandwidth-reference *value*

   The default bandwidth reference value is 100 Mbps.

Setting the maximum number of OSPFv3 ECMP routes

About this task

OSPFv3 might find multiple equal-cost routes to the same destination, which can be used to share the traffic load. This task allows you to set the maximum number of ECMP routes.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set the maximum number of ECMP routes.

   maximum load-balancing *number*

   The maximum number of OSPFv3 ECMP routes equals the maximum number of ECMP routes supported by the system.

Setting a preference for OSPFv3

About this task

A router can run multiple routing protocols. The system assigns a priority for each protocol. When these routing protocols find the same route, the route found by the protocol with the highest priority is selected.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

```
ospfv3 [ process-id | vpn-instance vpn-instance-name ] *
```

3. Set a preference for OSPFv3.

```
preference [ ase ] { preference | route-policy route-policy-name } *
```

By default, the preference of OSPFv3 internal routes is 10, and the preference of OSPFv3 external routes is 150.

Configuring OSPFv3 route redistribution

Restrictions and guidelines

Because OSPFv3 is a link state routing protocol, it cannot directly filter LSAs to be advertised. OSPFv3 filters only redistributed routes. Only routes that are not filtered out can be advertised in LSAs.

Procedure

1. Enter system view.

```
system-view
```

2. Enter OSPFv3 view.

```
ospfv3 [ process-id | vpn-instance vpn-instance-name ] *
```

3. Configure route redistribution.

```
import-route { direct | guard | static } [ cost cost-value | nssa-only |
route-policy route-policy-name | tag tag | type type ] *
```

```
import-route { isisv6 | ospfv3 | ripng } [ process-id | all-processes ]
[ allow-direct | cost cost-value | nssa-only | route-policy
route-policy-name | tag tag | type type ] *
```

By default, OSPFv3 does not redistribute routes.

The **import-route bgp4+** command redistributes only EBGP routes. The **import-route bgp4+ allow-ibgp** command redistributes both EBGP and IBGP routes, which might cause routing loops. There, use the **import-route bgp4+ allow-ibgp** command with caution.

4. (Optional.) Configure OSPFv3 to filter redistributed routes.

```
filter-policy { ipv6-acl-number | prefix-list prefix-list-name }
export [ bgp4+ | direct | { isisv6 | ospfv3 | ripng } [ process-id ] |
static ]
```

By default, OSPFv3 accepts all redistributed routes.

This command filters only routes redistributed by the **import-route** command. If no routes are redistributed by the **import-route** command, this command does not take effect.

5. Set a tag for redistributed routes.

```
default tag tag
```

By default, the tag of redistributed routes is 1.

Configuring default route redistribution

About this task

The **import-route** command cannot redistribute a default external route. To redistribute a default route, perform this task.

Procedure

1. Enter system view.

```
system-view
```

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Redistribute a default route.

   default-route-advertise [[**always** | **permit-calculate-other**] | **cost** *cost-value* | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] *

   By default, no default route is redistributed.

4. Set a tag for redistributed routes.

   default tag *tag*

   By default, the tag of redistributed routes is 1.

Setting OSPFv3 timers

Setting OSPFv3 packet timers

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the hello interval.

   ospfv3 timer hello *seconds* [**instance** *instance-id*]

   The default hello interval on P2P and broadcast interfaces is 10 seconds. The default hello interval on P2MP and NBMA interfaces is 30 seconds.

4. Set the dead interval.

   ospfv3 timer dead *seconds* [**instance** *instance-id*]

   The default dead interval on P2P and broadcast interfaces is 40 seconds. The default dead interval on P2MP and NBMA interfaces is 120 seconds.

   The dead interval set on neighboring interfaces cannot be too short. If the interval is too short, a neighbor is easily down.

5. Set the poll interval.

   ospfv3 timer poll *seconds* [**instance** *instance-id*]

   By default, the poll interval is 120 seconds.

6. Set the LSA retransmission interval.

   ospfv3 timer retransmit *interval* [**instance** *instance-id*]

   The default LSA retransmission interval is 5 seconds.

   The LSA retransmission interval cannot be too short. If the interval is too short, unnecessary retransmissions will occur.

Setting LSA transmission delay

About this task

Each LSA in the LSDB has an age that increases by 1 every second, but the age does not change during transmission. Therefore, it is necessary to add a transmission delay into the age time, especially for low-speed links.

Procedure

1. Enter system view.

system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the LSA transmission delay.

   ospfv3 trans-delay *seconds* [**instance** *instance-id*]

   By default, the LSA transmission delay is 1 second.

Setting SPF calculation interval

About this task

LSDB changes result in SPF calculations. When the topology changes frequently, a large amount of network and router resources are occupied by SPF calculation. You can adjust the SPF calculation interval to reduce the impact.

For a stable network, the minimum interval is used. If network changes become frequent, the SPF calculation interval increases by the incremental interval × 2^{n-2} for each calculation until the maximum interval is reached. The value *n* is the number of calculation times.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set the SPF calculation interval.

   spf-schedule-interval *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   By default, the maximum interval is 5 seconds, the minimum interval is 50 milliseconds, and the incremental interval is 200 milliseconds.

Setting the LSA generation interval

About this task

You can adjust the LSA generation interval to protect network resources and routers from being over consumed by frequent network changes.

For a stable network, the minimum interval is used. If network changes become frequent, the LSA generation interval increases by the incremental interval × 2^{n-2} for each generation until the maximum interval is reached. The value *n* is the number of generation times.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set the LSA generation interval.

   lsa-generation-interval *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   By default, the maximum interval is 5 seconds, the minimum interval is 0 milliseconds, and the incremental interval is 0 milliseconds.

Setting the LSU transmit rate

About this task

Sending large numbers of LSU packets affects router performance and consumes a large amount of network bandwidth. You can configure the router to send LSU packets at an interval and to limit the maximum number of LSU packets sent out of an OSPFv3 interface at each interval.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set the LSU transmit rate.

   transmit-pacing interval *interval* **count** *count*

   By default, an OSPFv3 interface sends a maximum of three LSU packets every 20 milliseconds.

Setting a DR priority for an interface

About this task

The router priority is used for DR election. Interfaces having the priority 0 cannot become a DR or BDR.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set a router priority.

   ospfv3 dr-priority *priority* [**instance** *instance-id*]

   The default router priority is 1.

Configuring OSPFv3 packet parameters

Ignoring MTU check for DD packets

About this task

When LSAs are few in DD packets, it is unnecessary to check the MTU in DD packets to improve efficiency.

Restrictions and guidelines

A neighbor relationship can be established only if the interface's MTU is the same as that of the peer.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Ignore MTU check for DD packets.

   ospfv3 mtu-ignore [**instance** *instance-id*]

   By default, OSPFv3 does not ignore MTU check for DD packets.

Disabling interfaces from receiving and sending OSPFv3 packets

About this task

After an OSPFv3 interface is set to **silent**, direct routes of the interface can still be advertised in Intra-Area-Prefix LSAs through other interfaces, but other OSPFv3 packets cannot be advertised. No neighboring relationship can be established on the interface. This feature can enhance the adaptability of OSPFv3 networking.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Disable interfaces from receiving and sending OSPFv3 packets.

   silent-interface { *interface-type interface-number* | **all** }

   By default, the interfaces can receive and send OSPFv3 packets.

   This command disables only the interfaces that run the current process. However, multiple OSPFv3 processes can disable the same interface from receiving and sending OSPFv3 packets.

Configuring prefix suppression

About prefix suppression

By default, an OSPFv3 interface advertises all of its prefixes in LSAs. To speed up OSPFv3 convergence, you can suppress interfaces from advertising all of their prefixes. This feature helps improve network security by preventing IP routing to the suppressed networks.

When prefix suppression is enabled:

- OSPFv3 does not advertise the prefixes of suppressed interfaces in Type-8 LSAs.
- On broadcast and NBMA networks, the DR does not advertise the prefixes of suppressed interfaces in Type-9 LSAs that reference Type-2 LSAs.
- On P2P and P2MP networks, OSPFv3 does not advertise the prefixes of suppressed interfaces in Type-9 LSAs that reference Type-1 LSAs.

Restrictions and guidelines for prefix suppression

As a best practice, configure prefix suppression on all OSPFv3 routers if you want to use prefix suppression.

Configuring prefix suppression for an OSPFv3 process

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enable prefix suppression for the OSPFv3 process.

   prefix-suppression

   By default, prefix suppression is disabled for an OSPFv3 process.

   Enabling prefix suppression for an OSPFv3 process does not suppress the prefixes of loopback interfaces and passive interfaces.

Configuring prefix suppression for an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable prefix suppression for the interface.

   ospfv3 prefix-suppression [**disable**] [**instance** *instance-id*]

   By default, prefix suppression is disabled for an interface.

Configuring a stub router

About this task

A stub router is used for traffic control. It reports its status as a stub router to neighboring OSPFv3 routers. The neighboring routers can have a route to the stub router, but they do not use the stub router to forward data.

Use either of the following methods to configure a router as a stub router:

- Clear the R-bit of the Option field in Type-1 LSAs. When the R-bit is clear, the OSPFv3 router can participate in OSPFv3 topology distribution without forwarding traffic.
- Use the OSPFv3 max-metric router LSA feature. This feature enables OSPFv3 to advertise its locally generated Type-1 LSAs with a maximum cost of 65535. Neighbors do not send packets to the stub router as long as they have a route with a smaller cost.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Configure the router as a stub router.

   - Configure the router as a stub router and clear the R-bit of the Option field in Type-1 LSAs.

     stub-router r-bit [**include-stub** | **on-startup** { *seconds* | **wait-for-bgp** [*seconds*] }] *

   - Configure the router as a stub router and advertise the locally generated Type-1 LSAs with the maximum cost of 65535.

stub-router max-metric [**external-lsa** [*max-metric-value*] | **summary-lsa** [*max-metric-value*] | **include-stub** | **on-startup** { *seconds* | **wait-for-bgp** [*seconds*] }] *

By default, the router is not configured as a stub router.

A stub router is not related to a stub area.

Configuring OSPFv3 GR

About OSPFv3 GR

GR ensures forwarding continuity when a routing protocol restarts or an active/standby switchover occurs.

Two routers are required to complete a GR process. The following are router roles in a GR process:

- **GR restarter**—Graceful restarting router. It must be Graceful Restart capable.
- **GR helper**—The neighbor of the GR restarter. It helps the GR restarter to complete the GR process.

To prevent service interruption after a master/backup switchover, a GR restarter running OSPFv3 must perform the following tasks:

- Keep the GR restarter forwarding entries stable during reboot.
- Establish all adjacencies and obtain complete topology information after reboot.

After the active/standby switchover, the GR restarter sends a Grace LSA to tell its neighbors that it performs a GR. Upon receiving the Grace LSA, the neighbors with the GR helper capability enter the helper mode (and are called GR helpers). Then, the GR restarter retrieves its adjacencies and LSDB with the help of the GR helpers.

Restrictions and guidelines for OSPFv3 GR

You cannot enable OSPFv3 NSR on a device that acts as GR restarter.

Configuring GR restarter

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enable the GR capability.

   graceful-restart enable [**global** | **planned-only**] *

   By default, OSPFv3 GR restarter capability is disabled.

4. (Optional.) Set the GR interval.

   graceful-restart interval *interval*

   By default, the GR interval is 120 seconds.

Configuring GR helper

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enable the GR helper capability.

   graceful-restart helper enable [**planned-only**]

   By default, the GR helper capability is enabled.

4. Enable strict LSA checking.

   graceful-restart helper strict-lsa-checking

   By default, strict LSA checking is disabled.

Triggering OSPFv3 GR

About this task

OSPFv3 GR is triggered by an active/standby switchover or when this task is performed.

Procedure

To trigger OSPFv3 GR, execute the **reset ospfv3** [*process-id*] **process graceful-restart** command in user view.

Configuring OSPFv3 NSR

About this task

Nonstop routing (NSR) backs up OSPFv3 link state information from the active process to the standby process. After an active/standby switchover, NSR can complete link state recovery and route regeneration without tearing down adjacencies or impacting forwarding services.

NSR does not require the cooperation of neighboring devices to recover routing information, and it is typically used more often than GR.

Restrictions and guidelines

A device that has OSPFv3 NSR enabled cannot act as GR restarter.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enable OSPFv3 NSR.

   non-stop-routing

   By default, OSPFv3 NSR is disabled.

   This command takes effect only for the current process. As a best practice, enable OSPFv3 NSR for each process if multiple OSPFv3 processes exist.

Configuring BFD for OSPFv3

About this task

Bidirectional forwarding detection (BFD) provides a mechanism to quickly detect the connectivity of links between OSPFv3 neighbors, improving the convergence speed of OSPFv3.For more information about BFD, see BFD configuration in *Network Management and Monitoring Configuration Guide*.

After discovering neighbors by sending hello packets, OSPFv3 notifies BFD of the neighbor addresses, and BFD uses these addresses to establish sessions. Before a BFD session is established, it is in the down state. In this state, BFD control packets are sent at an interval of no less than 1 second to reduce BFD control packet traffic. After the BFD session is established, BFD control packets are sent at the negotiated interval, thereby implementing fast fault detection.

To configure BFD for OSPFv3, you need to configure OSPFv3 first.

Procedure

1. Enter system view.

system-view

2. Enter OSPFv3 view.

ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Specify a router ID.

router-id *router-id*

4. Quit the OSPFv3 view.

quit

5. Enter interface view.

interface *interface-type interface-number*

6. Enable an OSPFv3 process on the interface.

ospfv3 *process-id* **area** *area-id* [**instance** *instance-id*]

7. Enable BFD on the interface.

ospfv3 bfd enable [**instance** *instance-id*]

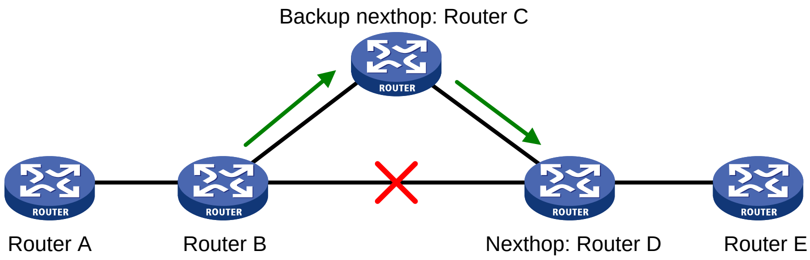
By default, BFD is disabled on the OSPFv3 interface.

Configuring OSPFv3 FRR

About OSPFv3 FRR

A primary link failure can cause packet loss and even a routing loop until OSPFv3 completes routing convergence based on the new network topology. OSPFv3 FRR enables fast rerouting to minimize the failover time.

Figure 1 Network diagram for OSPFv3 FRR



As shown in Figure 1, configure FRR on Router B. OSPFv3 FRR automatically calculates a backup next hop or specifies a backup next hop by using a routing policy. When the primary link fails, OSPFv3 directs packets to the backup next hop. At the same time, OSPFv3 calculates the shortest path based on the new network topology. It forwards packets over the path after network convergence.

You can configure OSPFv3 FRR to calculate a backup next hop by using the loop free alternate (LFA) algorithm, or specify a backup next hop by using a routing policy.

Configuring OSPFv3 FRR to use the LFA algorithm to calculate a backup next hop

Restrictions and guidelines

Do not use the **fast-reroute lfa** command together with the **vlink-peer** command.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. (Optional.) Disable LFA on an interface.

   ospfv3 fast-reroute lfa-backup exclude

   By default, the interface on which LFA is enabled can be selected as a backup interface.

4. Return to system view.

   quit

5. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

6. Enable OSPFv3 FRR to use the LFA algorithm to calculate a backup next hop.

   fast-reroute lfa [**abr-only**]

   By default, OSPFv3 FRR is disabled.

   If **abr-only** is specified, the route to the ABR is selected as the backup path.

Configuring OSPFv3 FRR to use a backup next hop in a routing policy

About this task

Before you perform this task, use the **apply ipv6 fast-reroute backup-interface** command to specify a backup next hop in the routing policy to be used. For more information about the **apply ipv6 fast-reroute backup-interface** command and routing policy configuration, see "Configuring routing policies."

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. (Optional.) Disable LFA on an interface.

   ospfv3 fast-reroute lfa-backup exclude

   By default, the interface is enabled with LFA and it can be selected as a backup interface.

4. Return to system view.

   quit

5. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

6. Configure OSPFv3 FRR to use a backup next hop in a routing policy.

   fast-reroute route-policy *route-policy-name*

   By default, OSPFv3 FRR is disabled.

Configuring BFD control packet mode for OSPFv3 FRR

About this task

By default, OSPFv3 FRR does not use BFD to detect primary link failures. To speed up OSPFv3 convergence, enable BFD control packet mode for OSPFv3 FRR to detect primary link failures. This mode requires BFD configuration on both OSPFv3 routers on the link.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for OSPFv3 FRR.

   ospfv3 primary-path-detect bfd ctrl [**instance** *instance-id*]

   By default, BFD control packet mode is disabled for OSPFv3 FRR.

Configuring BFD echo packet mode for OSPFv3 FRR

About this task

By default, OSPFv3 FRR does not use BFD to detect primary link failures. To speed up OSPFv3 convergence, enable BFD echo packet mode for OSPFv3 FRR to detect primary link failures. This mode requires BFD configuration on one OSPFv3 router on the link.

Procedure

1. Enter system view.

   system-view

2. Configure the source IPv6 address of BFD echo packets.

   bfd echo-source-ipv6 *ipv6-address*

   By default, the source IPv6 address of BFD echo packets is not configured.

   The source IPv6 address cannot be on the same network segment as any local interface's IP address.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD echo packet mode for OSPFv3 FRR.

   ospfv3 primary-path-detect bfd echo [**instance** *instance-id*]

   By default, BFD echo packet mode is disabled for OSPFv3 FRR.

Configuring OSPFv3 security features

Configuring OSPFv3 authentication

About this task

Perform this task to configure OSPFv3 area and interface authentication by using a keychain.

OSPFv3 adds the configured key into outgoing packets, and uses the key to authenticate incoming packets. Only the packets that pass the authentication can be received. If a packet fails the authentication, the OSPFv3 neighbor relationship cannot be established.

Restrictions and guidelines for OSPFv3 authentication

If you configure OSPFv3 authentication for both an area and an interface in that area, the interface uses the OSPFv3 authentication configured on it.

Configuring OSPFv3 area authentication

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enter OSPFv3 area view.

   area *area-id*

4. Configure area authentication.

   authentication-mode keychain *keychain-name*

   By default, no authentication is configured.

   For more information about keychains, see keychain configuration in *Security Configuration Guide*.

Configuring OSPF interface authentication

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure interface authentication.

   ospfv3 authentication-mode keychain *keychain-name* [**instance** *instance-id*]

   By default, no authentication is configured.

   For more information about keychains, see keychain configuration in *Security Configuration Guide*.

Applying an IPsec profile for authenticating OSPFv3 packets

About this task

To protect routing information and prevent attacks, you can configure OSPFv3 to authenticate protocol packets by using an IPsec profile.

An IPsec profile contains inbound and outbound security parameter indexes (SPIs). OSPFv3 compares the inbound SPI defined in the IPsec profile with the outbound SPI in the received packets.

Two OSPFv3 devices accept the packets from each other and establish a neighbor relationship only if the SPIs are the same and the relevant IPsec profiles match.

For more information about IPsec profiles, see *Security Configuration Guide*.

Restrictions and guidelines for applying an IPsec profile

You can configure an IPsec profile for an area, an interface, or a virtual link.

- To implement area-based IPsec protection, configure the same IPsec profile on the routers in the target area.
- To implement interface-based IPsec protection, configure the same IPsec profile on the interfaces between two neighboring routers.
- To implement virtual link-based IPsec protection, configure the same IPsec profile on the two routers connected over the virtual link.
- If an interface and its area each have an IPsec profile configured, the interface uses its own IPsec profile.
- If a virtual link and area 0 each have an IPsec profile configured, the virtual link uses its own IPsec profile.

Applying an IPsec profile to an area

1. Enter system view.
   ```
   system-view
   ```
2. Enter OSPFv3 view.
   ```
   ospfv3 [ process-id | vpn-instance vpn-instance-name ] *
   ```
3. Enter OSPFv3 area view.
   ```
   area area-id
   ```
4. Apply an IPsec profile to the area.
   ```
   enable ipsec-profile profile-name
   ```
   By default, no IPsec profile is applied.

Applying an IPsec profile to an interface

1. Enter system view.
   ```
   system-view
   ```
2. Enter interface view.
   ```
   interface interface-type interface-number
   ```
3. Apply an IPsec profile to the interface.
   ```
   ospfv3 ipsec-profile profile-name [ instance instance-id ]
   ```
   By default, no IPsec profile is applied.

Applying an IPsec profile to a virtual link

1. Enter system view.
   ```
   system-view
   ```
2. Enter OSPFv3 view.
   ```
   ospfv3 [ process-id | vpn-instance vpn-instance-name ] *
   ```
3. Enter OSPFv3 area view.
   ```
   area area-id
   ```
4. Apply an IPsec profile to a virtual link.
   ```
   vlink-peer router-id [ dead seconds | hello seconds | instance
   instance-id | ipsec-profile profile-name | retransmit seconds |
   trans-delay seconds ] *
   ```

By default, no IPsec profile is applied.

Configuring OSPFv3 logging and SNMP notifications

Enabling logging for neighbor state changes

About this task

With this feature enabled, the router delivers logs about neighbor state changes to its information center. The information center processes logs according to user-defined output rules (whether to output logs and where to output). For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Enable logging for neighbor state changes.

   log-peer-change

   By default, this feature is enabled.

Setting the maximum number of OSPFv3 logs

About this task

OSPFv3 logs include route calculation logs, neighbor logs, and LSA aging logs.

Procedure

1. Enter system view.

   system-view

2. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

3. Set the maximum number of OSPFv3 logs.

   event-log { **lsa-flush** | **peer** | **spf** } **size** *count*

   By default, the maximum number of LSA aging logs, neighbor logs, or route calculation logs is 10.

Configuring OSPFv3 network management

About this task

This task involves the following configurations:

- Bind an OSPFv3 process to MIB so that you can use network management software to manage the specified OSPFv3 process.
- Enable SNMP notifications for OSPFv3 to report important events.
- Set the SNMP notification output interval and the maximum number of SNMP notifications that can be output at each interval.

To report critical OSPFv3 events to an NMS, enable SNMP notifications for OSPFv3. For SNMP notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see the network management and monitoring configuration guide for the device.

The standard OSPFv3 MIB provides only single-instance MIB objects. To identify multiple OSPFv3 processes in the standard OSPFv3 MIB, you must assign a unique context name to each OSPFv3 process.

Context is a method introduced to SNMPv3 for multiple-instance management. For SNMPv1/v2c, you must specify a community name as a context name for protocol identification.

Procedure

1. Enter system view.

   system-view

2. Bind MIB to an OSPFv3 process.

   ospfv3 mib-binding *process-id*

   By default, MIB is bound to the process with the smallest process ID.

3. Enable SNMP notifications for OSPFv3.

   snmp-agent trap enable ospfv3 [**grrestarter-status-change** | **grhelper-status-change** | **if-state-change** | **if-cfg-error** | **if-bad-pkt** | **neighbor-state-change** | **nssatranslator-status-change** | **virtif-bad-pkt** | **virtif-cfg-error** | **virtif-state-change** | **virtgrhelper-status-change** | **virtneighbor-state-change**]*

   By default, SNMP notifications for OSPFv3 are enabled.

4. Enter OSPFv3 view.

   ospfv3 [*process-id* | **vpn-instance** *vpn-instance-name*] *

5. Configure an SNMP context for the OSPFv3 process.

   snmp context-name *context-name*

   By default, no SNMP context is configured for the OSPFv3 process.

6. (Optional.) Set the SNMP notification output interval and the maximum number of SNMP notifications that can be output at each interval.

   snmp trap rate-limit interval *trap-interval* **count** *trap-number*

   By default, OSPFv3 outputs a maximum of seven SNMP notifications within 10 seconds.

Display and maintenance commands for OSPFv3

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display summary route information on the OSPFv3 ABR.	**display ospfv3** [*process-id*] [**area** *area-id*] **abr-summary** [*ipv6-address prefix-length*] [**verbose**]
Display OSPFv3 neighbor information.	**display ospfv3** [*process-id*] [**area** *area-id*] **peer** [[*interface-type interface-number*] [**verbose**] | *peer-router-id* | **statistics**]
Display OSPFv3 request list information.	**display ospfv3** [*process-id*] [**area** *area-id*] **request-queue** [*interface-type interface-number*] [*neighbor-id*]
Display OSPFv3 retransmission list	**display ospfv3** [*process-id*] [**area** *area-id*]

Task	Command
information.	**retrans-queue** [*interface-type interface-number*] [*neighbor-id*]
Display OSPFv3 topology information.	**display ospfv3** [*process-id*] [**area** *area-id*] **spf-tree** [**verbose**]
Display OSPFv3 process information.	**display ospfv3** [*process-id*] [**verbose**]
Display information about the routes to OSPFv3 ABR and ASBR.	**display ospfv3** [*process-id*] **abr-asbr**
Display summary route information on the OSPFv3 ASBR.	**display ospfv3** [*process-id*] **asbr-summary** [*ipv6-address prefix-length*] [**verbose**]
Display OSPFv3 log information.	**display ospfv3** [*process-id*] **event-log** { **lsa-flush** | **peer** | **spf** }
Display OSPFv3 GR information.	**display ospfv3** [*process-id*] **graceful-restart** [**verbose**]
Display OSPFv3 interface information.	**display ospfv3** [*process-id*] **interface** [*interface-type interface-number* | **verbose**]
Display OSPFv3 LSDB information.	**display ospfv3** [*process-id*] **lsdb** [{ **external** | **grace** | **inter-prefix** | **inter-router** | **intra-prefix** | **link** | **network** | **nssa** | **router** | **unknown** [*type*] } [*link-state-id*] [**originate-router** *router-id* | **self-originate**] | **statistics** | **total** | **verbose**]
Display OSPFv3 next hop information.	**display ospfv3** [*process-id*] **nexthop**
Display OSPFv3 NSR information.	**display ospfv3** [*process-id*] **non-stop-routing**
Display OSPFv3 routing information.	**display ospfv3** [*process-id*] **routing** [*ipv6-address prefix-length*]
Display OSPFv3 statistics.	**display ospfv3** [*process-id*] **statistics** [**error**]
Display OSPFv3 virtual link information.	**display ospfv3** [*process-id*] **vlink**
Clear OSPFv3 log information.	**reset ospfv3** [*process-id*] **event-log** [**lsa-flush** | **peer** | **spf**]
Restart an OSPFv3 process.	**reset ospfv3** [*process-id*] **process** [**graceful-restart**]
Restart OSPFv3 route redistribution.	**reset ospfv3** [*process-id*] **redistribution**
Clear OSPFv3 statistics.	**reset ospfv3** [*process-id*] **statistics**

OSPFv3 configuration examples

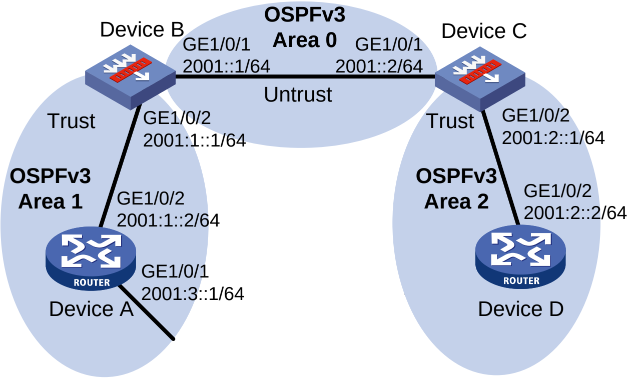
Example: Configuring OSPFv3 stub area

Network configuration

As shown in Figure 2:

- Enable OSPFv3 on all devices.
- Split the AS into three areas.
- Configure Device B and Device C as ABRs to forward routing information between areas.
- Configure Area 2 as a stub area to reduce LSAs in the area without affecting route reachability.

Figure 2 Network diagram



Procedure

# Configure Device B.

1. Configure IPv6 addresses for interfaces correctly according to Figure 2.
   a. Configure an IP address for GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2001::1 64
[DeviceB-GigabitEthernet1/0/1] quit
```

   b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device B to a security zone.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.
   a. Configure a security policy to allow OSPFv3 neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **ospflocalin** and permit Device B to receive OSPFv3 packets from Device C.

```
[DeviceB] security-policy ipv6
```

```
[DeviceB-security-policy-ipv6] rule name ospflocalin
[DeviceB-security-policy-ipv6-0-ospflocalin] source-zone untrust
[DeviceB-security-policy-ipv6-0-ospflocalin] destination-zone local
[DeviceB-security-policy-ipv6-0-ospflocalin] service ospf
[DeviceB-security-policy-ipv6-0-ospflocalin] action pass
[DeviceB-security-policy-ipv6-0-ospflocalin] quit
```

# Create security policy rule **ospflocalout** and permit Device B to send OSPFv3 packets to Device C.

```
[DeviceB-security-policy-ipv6] rule name ospflocalout
[DeviceB-security-policy-ipv6-1-ospflocalout] source-zone local
[DeviceB-security-policy-ipv6-1-ospflocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-ospflocalout] service ospf
[DeviceB-security-policy-ipv6-1-ospflocalout] action pass
[DeviceB-security-policy-ipv6-1-ospflocalout] quit
```

b. Configure a security policy to allow OSPFv3 neighbor relationship establishment by permitting traffic between security zone **trust** and security zone **local**.

# Create security policy rule **trust-local** and permit Device A to send OSPFv3 packets to Device B.

```
[DeviceB-security-policy-ipv6] rule name trust-local
[DeviceB-security-policy-ipv6-2-trust-local] source-zone trust
[DeviceB-security-policy-ipv6-2-trust-local] destination-zone local
[DeviceB-security-policy-ipv6-2-trust-local] service ospf
[DeviceB-security-policy-ipv6-2-trust-local] action pass
[DeviceB-security-policy-ipv6-2-trust-local] quit
```

# Create security policy rule **local-trust** and permit Device B to send OSPFv3 packets to Device A.

```
[DeviceB-security-policy-ip] rule name local-trust
[DeviceB-security-policy-ipv6-3-local-trust] source-zone local
[DeviceB-security-policy-ipv6-3-local-trust] destination-zone trust
[DeviceB-security-policy-ipv6-3-local-trust] service ospf
[DeviceB-security-policy-ipv6-3-local-trust] action pass
[DeviceB-security-policy-ipv6-3-local-trust] quit
```

c. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust**.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-4-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-4-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-4-trust-untrust] source-ip-subnet 2001:1:: 64
[DeviceB-security-policy-ipv6-4-trust-untrust] destination-ip-subnet 2001:2::
64
[DeviceB-security-policy-ipv6-4-trust-untrust] action pass
[DeviceB-security-policy-ipv6-4-trust-untrust] quit
```

# Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceB-security-policy-ipv6] rule name untrust-trust
[DeviceB-security-policy-ipv6-5-untrust-trust] source-zone untrust
[DeviceB-security-policy-ipv6-5-untrust-trust] destination-zone trust
```

```
[DeviceB-security-policy-ipv6-5-untrust-trust] source-ip-subnet 2001:2:: 64
[DeviceB-security-policy-ipv6-5-untrust-trust] destination-ip-subnet 2001:1::
64
[DeviceB-security-policy-ipv6-5-untrust-trust] service ospf
[DeviceB-security-policy-ipv6-5-untrust-trust] quit
[DeviceB-security-policy-ipv6] quit
```

# Configure Device C.

1. Configure IPv6 addresses for interfaces correctly according to Figure 2.
   a. Configure an IP address for GigabitEthernet 1/0/1.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ipv6 address 2001::2 64
[DeviceC-GigabitEthernet1/0/1] quit
```

   b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device C to a security zone.

```
[DeviceC] security-zone name untrust
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceC-security-zone-Untrust] quit
[DeviceC] security-zone name trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
```

3. Configure security policies.
   a. Configure a security policy to allow OSPFv3 neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **ospflocalin** and permit Device C to receive OSPFv3 packets from Device B.

```
[DeviceC] security-policy ipv6
[DeviceC-security-policy-ipv6] rule name ospflocalin
[DeviceC-security-policy-ipv6-0-ospflocalin] source-zone untrust
[DeviceC-security-policy-ipv6-0-ospflocalin] destination-zone local
[DeviceC-security-policy-ipv6-0-ospflocalin] service ospf
[DeviceC-security-policy-ipv6-0-ospflocalin] action pass
[DeviceC-security-policy-ipv6-0-ospflocalin] quit
```

   # Create security policy rule **ospflocalout** and permit Device C to send OSPFv3 packets to Device B.

```
[DeviceC-security-policy-ipv6] rule name ospflocalout
[DeviceC-security-policy-ipv6-1-ospflocalout] source-zone local
[DeviceC-security-policy-ipv6-1-ospflocalout] destination-zone untrust
[DeviceC-security-policy-ipv6-1-ospflocalout] service ospf
[DeviceC-security-policy-ipv6-1-ospflocalout] action pass
[DeviceC-security-policy-ipv6-1-ospflocalout] quit
```

   b. Configure a security policy to allow OSPFv3 neighbor relationship establishment by permitting traffic between security zone **trust** and security zone **local**.

   # Create security policy rule **trust-local** and permit Device D to send OSPFv3 packets to Device C.

```
[DeviceC-security-policy-ipv6] rule name trust-local
[DeviceC-security-policy-ipv6-2-trust-local] source-zone trust
```

```
[DeviceC-security-policy-ipv6-2-trust-local] destination-zone local
[DeviceC-security-policy-ipv6-2-trust-local] service ospf
[DeviceC-security-policy-ipv6-2-trust-local] action pass
[DeviceC-security-policy-ipv6-2-trust-local] quit
```

# Create security policy rule **local-trust** and permit Device C to send OSPFv3 packets to Device D.

```
[DeviceC-security-policy-ipv6] rule name local-trust
[DeviceC-security-policy-ipv6-3-local-trust] source-zone local
[DeviceC-security-policy-ipv6-3-local-trust] destination-zone trust
[DeviceC-security-policy-ipv6-3-local-trust] service ospf
[DeviceC-security-policy-ipv6-3-local-trust] action pass
[DeviceC-security-policy-ipv6-3-local-trust] quit
```

c. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust**.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceC-security-policy-ipv6] rule name trust-untrust
[DeviceC-security-policy-ipv6-4-trust-untrust] source-zone trust
[DeviceC-security-policy-ipv6-4-trust-untrust] destination-zone untrust
[DeviceC-security-policy-ipv6-4-trust-untrust] source-ip-subnet 2001:2:: 64
[DeviceC-security-policy-ipv6-4-trust-untrust] destination-ip-subnet 2001:1::
64
[DeviceC-security-policy-ipv6-4-trust-untrust] action pass
[DeviceC-security-policy-ipv6-4-trust-untrust] quit
```

# Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceC-security-policy-ipv6] rule name untrust-trust
[DeviceC-security-policy-ipv6-5-untrust-trust] source-zone untrust
[DeviceC-security-policy-ipv6-5-untrust-trust] destination-zone trust
[DeviceC-security-policy-ipv6-5-untrust-trust] source-ip-subnet 2001:1:: 64
[DeviceC-security-policy-ipv6-5-untrust-trust] destination-ip-subnet 2001:2::
64
[DeviceC-security-policy-ipv6-5-trust-untrust] action pass
[DeviceC-security-policy-ipv6-5-trust-untrust] quit
[DeviceC-security-policy-ipv6] quit
```

4. Configure basic OSPFv3 functions:

# On Device A, enable OSPFv3 and specify the device ID as 1.1.1.1.

```
<DeviceA> system-view
[DeviceA] ospfv3 1
[DeviceA-ospfv3-1] router-id 1.1.1.1
[DeviceA-ospfv3-1] quit
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ospfv3 1 area 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] ospfv3 1 area 1
[DeviceA-GigabitEthernet1/0/2] quit
```

# On Device B, enable OSPFv3 and specify the device ID as 2.2.2.2.

```
[DeviceB] ospfv3 1
```

```
[DeviceB-ospfv3-1] router-id 2.2.2.2
[DeviceB-ospfv3-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ospfv3 1 area 0
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ospfv3 1 area 1
[DeviceB-GigabitEthernet1/0/2] quit
```

# On Device C, enable OSPFv3 and specify the device ID as 3.3.3.3.

```
[DeviceC] ospfv3 1
[DeviceC-ospfv3-1] router-id 3.3.3.3
[DeviceC-ospfv3-1] quit
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ospfv3 1 area 0
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] ospfv3 1 area 2
[DeviceC-GigabitEthernet1/0/2] quit
```

# On Device D, enable OSPFv3 and specify the device ID as 4.4.4.4.

```
<DeviceD> system-view
[DeviceD] ospfv3 1
[DeviceD-ospfv3-1] router-id 4.4.4.4
[DeviceD-ospfv3-1] quit
[DeviceD] interface gigabitethernet 1/0/2
[DeviceD-GigabitEthernet1/0/2] ospfv3 1 area 2
[DeviceD-GigabitEthernet1/0/2] quit
```

5. Configure Area 2 as a stub area:

   # Configure Device D.

```
[DeviceD] ospfv3
[DeviceD-ospfv3-1] area 2
[DeviceD-ospfv3-1-area-0.0.0.2] stub
[DeviceD-ospfv3-1-area-0.0.0.2] quit
[DeviceD-ospfv3-1] quit
```

   # Configure Device C, and specify the cost of the default route sent to the stub area as 10.

```
[DeviceC] ospfv3
[DeviceC-ospfv3-1] area 2
[DeviceC-ospfv3-1-area-0.0.0.2] stub
[DeviceC-ospfv3-1-area-0.0.0.2] default-cost 10
```

Verifying the configuration

# Display OSPFv3 routing table on Device D.

```
[DeviceD] display ospfv3 routing

               OSPFv3 Process 1 with Router ID 4.4.4.4
-------------------------------------------------------------------------
 I  - Intra area route,  E1 - Type 1 external route,  N1 - Type 1 NSSA route
 IA - Inter area route,  E2 - Type 2 external route,  N2 - Type 2 NSSA route
 *  - Selected route
```

```
*Destination: ::/0
 Type       : IA                        Area       : 0.0.0.2
 AdvRouter  : 1.1.1.1                   Preference : 10
 NibID      : 0x23000002                Cost       : 11
 Interface  : GE1/0/2                   BkInterface: N/A
 Nexthop    : FE80::CCA:73FF:FEA2:206
 BkNexthop  : N/A

*Destination: 2001::/64
 Type       : IA                        Area       : 0.0.0.2
 AdvRouter  : 1.1.1.1                   Preference : 10
 NibID      : 0x23000002                Cost       : 2
 Interface  : GE1/0/2                   BkInterface: N/A
 Nexthop    : FE80::CCA:73FF:FEA2:206
 BkNexthop  : N/A

*Destination: 2001:1::/64
 Type       : IA                        Area       : 0.0.0.2
 AdvRouter  : 1.1.1.1                   Preference : 10
 NibID      : 0x23000002                Cost       : 3
 Interface  : GE1/0/2                   BkInterface: N/A
 Nexthop    : FE80::CCA:73FF:FEA2:206
 BkNexthop  : N/A

*Destination: 2001:2::/64
 Type       : I                         Area       : 0.0.0.2
 AdvRouter  : 4.4.4.4                   Preference : 10
 NibID      : 0x23000001                Cost       : 1
 Interface  : GE1/0/2                   BkInterface: N/A
 Nexthop    : ::
 BkNexthop  : N/A

*Destination: 2001:3::/64
 Type       : IA                        Area       : 0.0.0.2
 AdvRouter  : 1.1.1.1                   Preference : 10
 NibID      : 0x23000002                Cost       : 4
 Interface  : GE1/0/2                   BkInterface: N/A
 Nexthop    : FE80::CCA:73FF:FEA2:206
 BkNexthop  : N/A

Total: 5
Intra area: 1         Inter area: 4         ASE: 0         NSSA: 0
```

# Verify that devices in area 1 can ping devices in area 2, and vice versa.

Contents

Configuring IS-IS

About IS-IS

IS-IS is an IGP used within an AS. It uses the SPF algorithm for route calculation.

Terminology

- **Intermediate system**—Similar to a router in TCP/IP, IS is the basic unit used in an IS-IS routing domain to generate and propagate routing information. Throughout this chapter, an IS refers to a router.
- **End system**—Similar to a host in TCP/IP, an ES does not run IS-IS. ISO defines the ES-IS protocol for communication between an ES and an IS.
- **Routing domain**—An RD comprises a group of ISs that exchange routing information with each other by using the same routing protocol.
- **Area**—An IS-IS routing domain can be split into multiple areas.
- **Link State Database**—All link states in the network form the LSDB. Each IS has a minimum of one LSDB. An IS uses the SPF algorithm and LSDB to generate IS-IS routes.
- **Link State Protocol Data Unit or Link State Packet**—An IS advertises link state information in an LSP.
- **Network Protocol Data Unit**—An NPDU is a network layer protocol packet in OSI, similar to an IP packet in TCP/IP.
- **Designated IS**—A DIS is elected on a broadcast network.
- **Network service access point**—An NSAP is an OSI network layer address. The NSAP identifies an abstract network service access point and describes the network address format in the OSI reference model.

IS-IS address

As shown in Figure 1, an NSAP address comprises the Initial Domain Part (IDP) and the Domain Specific Part (DSP). The IDP is analogous to the network ID of an IP address, and the DSP is analogous to the subnet and host ID.

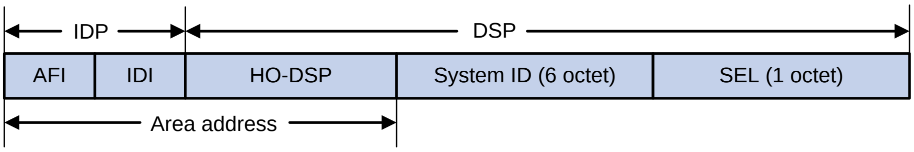
The IDP includes the Authority and Format Identifier (AFI) and the Initial Domain Identifier (IDI).

The DSP includes:

- **High Order Part of DSP (HO-DSP)**—Identifies the area.
- **System ID**—Identifies the host.
- **SEL**—Also known as the N-SEL or the NSAP selector (SEL). It is similar to the protocol identifier in IP and is used to identify the type of service. Different transport layer protocols correspond to different SELs.

The IDP and DSP are variable in length. The length of an NSAP address is in the range of 8 to 20 bytes.

Figure 1 NSAP address format



An IS-IS address contains the following components:

- Area address

  The area address comprises the IDP and the HO-DSP of the DSP, which identify the area and the routing domain. Different routing domains cannot have the same area address.

  Typically, a router only needs one area address, and all nodes in the same area must have the same area address. To support smooth area merging, partitioning, and switching, a router can have a maximum of three area addresses.

- System ID

  A system ID uniquely identifies a host or router. It has a fixed length of 48 bits (6 bytes).

  The system ID of a device can be generated from the router ID. For example, suppose a router uses the IP address 168.10.1.1 of Loopback 0 as the router ID. The system ID can be obtained in the following steps:

  a. Extend each decimal number of the IP address to three digits by adding 0s from the left, such as 168.010.001.001.

  b. Divide the extended IP address into three sections that each has four digits to get the system ID 1680.1000.1001.

  If you use other methods to define a system ID, make sure that it can uniquely identify the host or router.

- SEL

  An SEL is used to identify the type of service. Different transport layer protocols correspond to different SELs. It has a fixed length of 8 bits. All SELs in IP are 00.

NET

A network entity title (NET) identifies the network layer information of an IS. It does not include transport layer information. A NET is a special NSAP address with the SEL being 0. The length of a NET is in the range of 8 to 20 bytes, same as a NSAP address.

A NET includes the following parts:

- **Area ID**—Has a length of 1 to 13 bytes.
- **System ID**—A system ID uniquely identifies a host or router in the area and has a fixed length of 6 bytes.
- **SEL**—Has a value of 0 and a fixed length of 1 byte.

For example, for a NET ab.cdef.1234.5678.9abc.00, the area ID is ab.cdef, the system ID is 1234.5678.9abc, and the SEL is 00.

Typically, a router only needs one NET, but it can have a maximum of three NETs for smooth area merging and partitioning. When you configure multiple NETs, make sure the system IDs are the same.

IS-IS area

IS-IS has a 2-level hierarchy to support large-scale networks. A large-scale routing domain is divided into multiple areas. Typically, a Level-1 router is deployed within an area. A Level-2 router is deployed between areas. A Level-1-2 router is deployed between Level-1 and Level-2 routers.

Level-1 router

A Level-1 router establishes neighbor relationships with Level-1 and Level-1-2 routers in the same area. It maintains an LSDB comprising intra-area routing information. A Level-1 router forwards packets destined for external areas to the nearest Level-1-2 router. Level-1 routers in different areas cannot establish neighbor relationships.

Level-2 router

A Level-2 router establishes neighbor relationships with Level-2 and Level-1-2 routers in the same area or in different areas. It maintains a Level-2 LSDB containing inter-area routing information. All the Level-2 and Level-1-2 routers must be contiguous to form the backbone of the IS-IS routing domain. Level-2 routers can establish neighbor relationships even if they are in different areas.

Level-1-2 router

A router with both Level-1 and Level-2 router functions is a Level-1-2 router. It can establish Level-1 neighbor relationships with Level-1 and Level-1-2 routers in the same area. It can establish Level-2 neighbor relationships with Level-2 and Level-1-2 routers in different areas. A Level-1 router can reach other areas only through a Level-1-2 router. The Level-1-2 router maintains two LSDBs, a Level-1 LSDB for intra-area routing and a Level-2 LSDB for inter-area routing.

IS-IS topology

Figure 2 shows one IS-IS network topology. Area 1 is the backbone that comprises a set of Level-2 routers. The other four areas are non-backbone areas connected to the backbone through Level-1-2 routers.

Figure 2 IS-IS topology 1

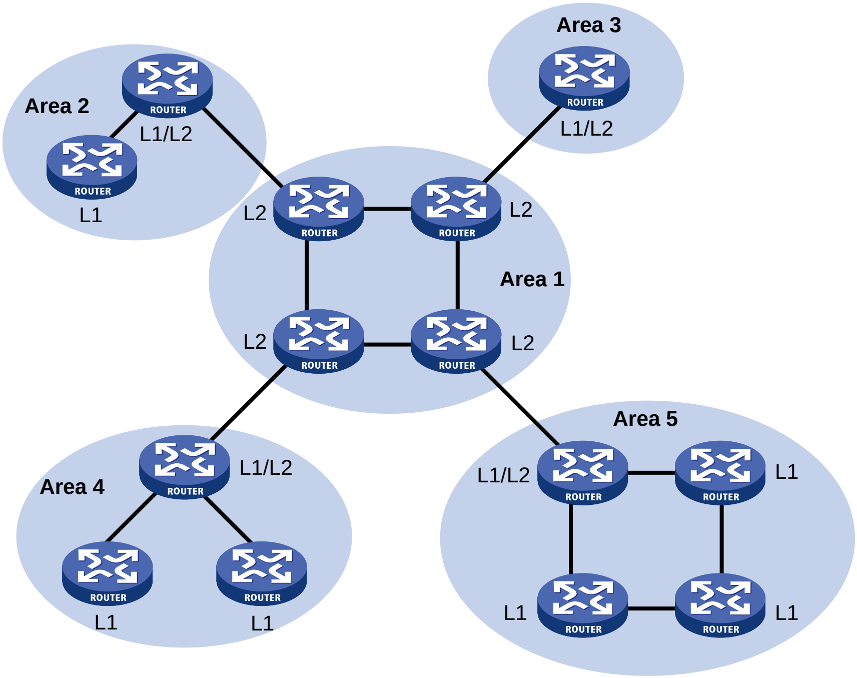
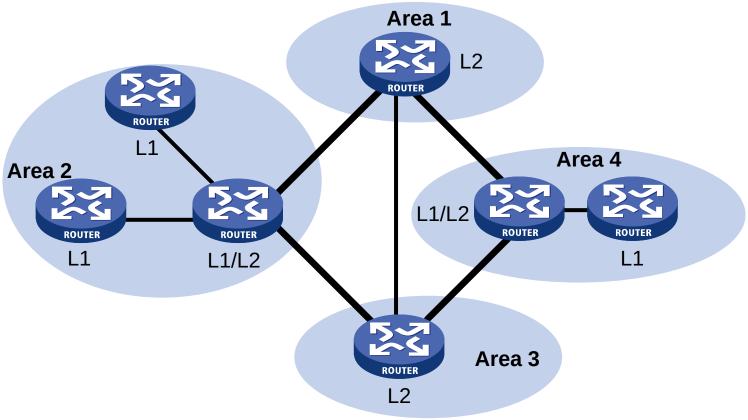


Figure 3 shows another IS-IS topology. No area is defined as the backbone in this topology. The backbone comprises all contiguous Level-2 and Level-1-2 routers in different areas. The IS-IS backbone does not need to be a specific area.

Figure 3 IS-IS topology 2



Both the Level-1 and Level-2 routers use the SPF algorithm to generate the shortest path tree.

Route leaking

Level-2 and Level-1-2 routers form a Level-2 area. An IS-IS routing domain comprises only one Level-2 area and multiple Level-1 areas. A Level-1 area must connect to the Level-1-2 area rather than another Level-1 area.

Level-1-2 routers send the routing information of Level-1 areas to the Level-2 area. Level-2 routers know the routing information of the entire IS-IS routing domain. By default, a Level-2 router does not advertise the routing information of other areas to a Level-1 area. A Level-1 router simply sends packets destined for other areas to the nearest Level-1-2 router. The path passing through the Level-1-2 router might not be the best. To solve this problem, IS-IS provides the route leaking feature.

Route leaking enables a Level-1-2 router to advertise the routes of other areas to the connected Level-1 area so that the Level-1 routers can select the optimal routes.

IS-IS network types

IS-IS supports broadcast networks (for example, Ethernet and Token Ring) and point-to-point networks (for example, PPP).

IS-IS cannot run on P2MP links.

DIS and pseudonodes

IS-IS routers on a broadcast network must elect a DIS.

The Level-1 and Level-2 DISs are elected separately. You can assign different priorities to a router for different level DIS elections. The higher the router priority, the more likely the router becomes the DIS. If multiple routers with the same highest DIS priority exist, the one with the highest Subnetwork Point of Attachment (SNPA) address will be elected. On a broadcast network, the SNPA address is the MAC address. A router can be the DIS for different levels.

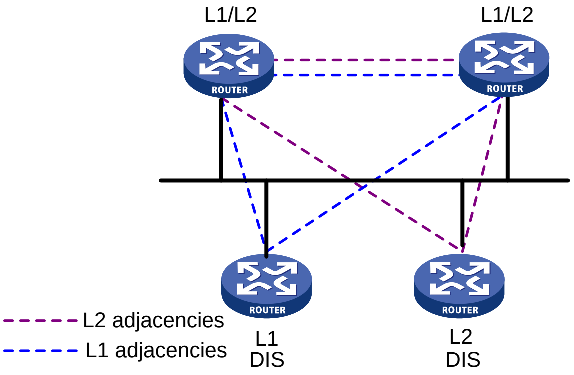
IS-IS DIS election differs from OSPF DIS election in the following ways:

- A router with priority 0 can also participate in the DIS election.

- When a router with a higher priority is added to the network, an LSP flooding process is performed to elect the router as the new DIS.

As shown in Figure 4, the same level routers on a network, including non-DIS routers, establish adjacency with each other.

Figure 4 DIS in the IS-IS broadcast network



The DIS creates and updates pseudonodes, and generates LSPs for the pseudonodes, to describe all routers on the network.

A pseudonode represents a virtual node on the broadcast network. It is not a real router. In IS-IS, it is identified by the system ID of the DIS and a 1-byte Circuit ID (a non-zero value).

Using pseudonodes simplifies network topology and can reduce the amount of resources consumed by SPF.

NOTE:

On an IS-IS broadcast network, all routers establish adjacency relationships, but they synchronize their LSDBs through the DIS.

IS-IS PDUs

PDU

IS-IS PDUs are encapsulated into link layer frames. An IS-IS PDU has two parts, the headers and the variable length fields. The headers comprise the PDU common header and the PDU specific header. All PDUs have the same PDU common header. The specific headers vary by PDU type.

Figure 5 PDU format

PDU common header	PDU specific header	Variable length fields (CLV)

Table 1 PDU types

Type	PDU Type	Acronym
15	Level-1 LAN IS-IS hello PDU	L1 LAN IIH
16	Level-2 LAN IS-IS hello PDU	L2 LAN IIH
17	Point-to-Point IS-IS hello PDU	P2P IIH
18	Level-1 Link State PDU	L1 LSP
20	Level-2 Link State PDU	L2 LSP
24	Level-1 Complete Sequence Numbers PDU	L1 CSNP

Type	PDU Type	Acronym
25	Level-2 Complete Sequence Numbers PDU	L2 CSNP
26	Level-1 Partial Sequence Numbers PDU	L1 PSNP
27	Level-2 Partial Sequence Numbers PDU	L2 PSNP

Hello PDU

IS-to-IS hello (IIH) PDUs are used by routers to establish and maintain neighbor relationships. On broadcast networks, Level-1 routers use Level-1 LAN IIHs, and Level-2 routers use Level-2 LAN IIHs. The P2P IIHs are used on point-to-point networks.

LSP

The LSPs carry link state information. LSPs include Level-1 LSPs and Level-2 LSPs. The Level-2 LSPs are sent by the Level-2 routers, and the Level-1 LSPs are sent by the Level-1 routers. The Level-1-2 router can send both types of LSPs.

SNP

A sequence number PDU (SNP) describes the complete or partial LSPs for LSDB synchronization.

SNPs include CSNP and PSNP, which are further divided into Level-1 CSNP, Level-2 CSNP, Level-1 PSNP, and Level-2 PSNP.

A CSNP describes the summary of all LSPs for LSDB synchronization between neighboring routers. On broadcast networks, CSNPs are sent by the DIS periodically (every 10 seconds by default). On point-to-point networks, CSNPs are sent only during the first adjacency establishment.

A PSNP only contains the sequence numbers of one or multiple latest received LSPs. It can acknowledge multiple LSPs at one time. When LSDBs are not synchronized, a PSNP is used to request missing LSPs from a neighbor.

CLV

The variable fields of PDU comprise multiple Code-Length-Value (CLV) triplets.

Figure 6 CLV format

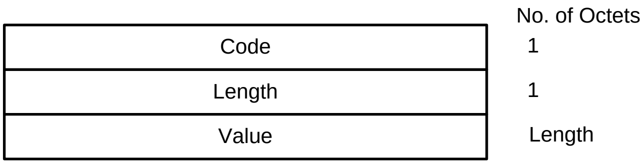


Table 2 shows that different PDUs contain different CLVs. Codes 1 through 10 are defined in ISO 10589 (code 3 and 5 are not shown in the table). Codes 128 through 132 are defined in RFC 1195. Codes 222 through 237 are defined in RFC 5120.

Table 2 CLV codes and PDU types

CLV Code	Name	PDU Type
1	Area Addresses	IIH, LSP
2	IS Neighbors (LSP)	LSP
4	Partition Designated Level 2 IS	L2 LSP
6	IS Neighbors (MAC Address)	LAN IIH
7	IS Neighbors (SNPA Address)	LAN IIH
8	Padding	IIH

CLV Code	Name	PDU Type
9	LSP Entries	SNP
10	Authentication Information	IIH, LSP, SNP
128	IP Internal Reachability Information	LSP
129	Protocols Supported	IIH, LSP
130	IP External Reachability Information	L2 LSP
131	Inter-Domain Routing Protocol Information	L2 LSP
132	IP Interface Address	IIH, LSP
222	MT-ISN	LSP
229	M-Topologies	IIH, LSP
235	MT IP. Reach	LSP
237	MT IPv6 IP. Reach	LSP

IPv6 IS-IS

IS-IS supports multiple network protocols, including IPv6. To support IPv6, the IETF added two type-length-values (TLVs) and a new network layer protocol identifier (NLPID).

The TLVs are as follows:

- **IPv6 Reachability**—Contains routing prefix and metric information to describe network reachability and has a type value of 236 (0xEC).
- **IPv6 Interface Address**—Same as the "IP Interface Address" TLV in IPv4 ISIS, except that the 32-bit IPv4 address is translated to the 128-bit IPv6 address.

The new NLPID is an 8-bit field that identifies which network layer protocol is supported. For IPv6, the NLPID is 142 (0x8E).

Protocols and standards

- ISO 8348, *Ad2 Network Services Access Points*
- ISO 9542, *ES-IS Routing Protocol*
- ISO 10589, *ISO IS-IS Routing Protocol*
- RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*
- RFC 2973, *IS-IS Mesh Groups*
- RFC 3277, *IS-IS Transient Blackhole Avoidance*
- RFC 3358, *Optional Checksums in ISIS*
- RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*
- RFC 3563, *Cooperative Agreement Between the ISOC/IETF and ISO/IEC Joint Technical Committee 1/Sub Committee 6 (JTC1/SC6) on IS-IS Routing Protocol Development*
- RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*
- RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*

- RFC 4444, *Management Information Base for Intermediate System to Intermediate System (IS-IS)*
- RFC 5029, *Definition of an IS-IS Link Attribute Sub-TLV*
- RFC 5089, *IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery*
- RFC 5120, *Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)*
- RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*
- RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*
- RFC 5302, *Domain-Wide Prefix Distribution with Two-Level IS-IS*
- RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*
- RFC 5304, *IS-IS Cryptographic Authentication*
- RFC 5305, *IS-IS Extensions for Traffic Engineering*
- RFC 5306, *Restart Signaling for IS-IS*
- RFC 5308, *Routing IPv6 with IS-IS*
- RFC 5310, *IS-IS Generic Cryptographic Authentication*
- RFC 5311, *Simplified Extension of Link State PDU (LSP) Space for IS-IS*
- RFC 6165, *Extensions to IS-IS for Layer-2 Systems*
- RFC 6213, *IS-IS BFD-Enabled TLV*
- RFC 6232, *Purge Originator Identification TLV for IS-IS*
- RFC 6233, *IS-IS Registry Extension for Purges*
- RFC 6329, *IS-IS Extensions Supporting IEEE 802.1aq Shortest Path Bridging*
- RFC 6571, *Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks*
- RFC 6823, *Advertising Generic Information in IS-IS*
- RFC 7142, *OSI IS-IS Intra-domain Routing Protocol*
- RFC 7356, *IS-IS Flooding Scope Link State PDUs (LSPs)*
- RFC 7370, *Updates to the IS-IS TLV Codepoints Registry*
- RFC 7602, *IS-IS Extended Sequence Number TLV*
- RFC 7645, *The Keying and Authentication for Routing Protocol (KARP) IS-IS Security Analysis*
- RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*
- RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability*
- RFC 7810, *IS-IS Traffic Engineering (TE) Metric Extensions*
- RFC 7813, *IS-IS Path Control and Reservation*
- RFC 7917, *Advertising Node Administrative Tags in IS-IS*
- RFC 7981, *IS-IS Extensions for Advertising Router Information*
- RFC 7987, *IS-IS Minimum Remaining Lifetime*

IPv4 IS-IS tasks at a glance

To configure IPv4 IS-IS, perform the following tasks:

1. Configuring basic IS-IS
   a. Enabling IPv4 IS-IS
   b. (Optional.) Setting the IS level and circuit level
   c. (Optional.) Configuring P2P network type for an interface

2. (Optional.) Configuring IS-IS route control
    - Configuring IS-IS link cost
    - Specifying a preference for IS-IS
    - Configuring the maximum number of ECMP routes
    - Configuring IS-IS route summarization
    - Advertising a default route
    - Configuring IS-IS route redistribution
    - Filtering routes calculated from received LSPs
    - Filtering redistributed routes
    - Configuring IS-IS route leaking
    - Advertising IS-IS link state information to BGP
3. (Optional.) Configuring IS-IS timers
    - Specifying the interval for sending IS-IS hello packets
    - Specifying the interval for sending IS-IS CSNP packets
    - Setting the maximum age of LSPs
    - Setting the LSP refresh interval and generation interval
    - Setting LSP sending intervals
    - Setting the SPF calculation interval
4. (Optional.) Configuring IS-IS packet-related features
    - Configuring a DIS priority for an interface
    - Configuring the tag value for an interface
    - Specifying the IS-IS hello multiplier
    - Disabling an interface from sending/receiving IS-IS packets
    - Enabling an interface to send small hello packets
    - Setting LSP lengths
    - Enabling LSP flash flooding
    - Enabling LSP fragment extension
5. (Optional.) Configuring advanced IS-IS features
    - Enabling source address check for hello packets on a P2P interface
    - Configuring convergence priorities for specific routes
    - Setting the LSDB overload bit
    - Configuring the ATT bit
    - Configuring system ID to host name mappings
    - Enabling IS-IS to group ECMP routes
6. (Optional.) Configuring IS-IS logging and SNMP notifications
    - Enabling the logging of neighbor state changes
    - Configuring IS-IS network management
7. (Optional.) Configuring IS-IS fast convergence
    - Enabling ISPF
    - Enabling prefix suppression
    - Configuring IS-IS PIC
8. (Optional.) Enhancing IS-IS network security
    - Configuring neighbor relationship authentication
    - Configuring area authentication

- Configuring routing domain authentication

9. (Optional.) Enhancing IS-IS network reliability
   - Configuring IS-IS GR
   - Configuring IS-IS NSR
   - Configuring BFD for IS-IS
   - Controlling adjacency establishment and maintenance based on BFD session state
   - Configuring IS-IS FRR

IPv6 IS-IS tasks at a glance

To configure IPv6 IS-IS, perform the following tasks:

1. Configuring basic IS-IS
   a. Enabling IPv6 IS-IS
   b. (Optional.) Setting the IS level and circuit level
   c. (Optional.) Configuring P2P network type for an interface
2. (Optional.) Configuring IPv6 IS-IS MTR
3. (Optional.) Configuring IS-IS route control
   - Configuring IS-IS link cost
   - Specifying a preference for IS-IS
   - Configuring the maximum number of ECMP routes
   - Configuring IS-IS route summarization
   - Advertising a default route
   - Configuring IS-IS route redistribution
   - Filtering routes calculated from received LSPs
   - Filtering redistributed routes
   - Configuring IS-IS route leaking
   - Advertising IS-IS link state information to BGP
4. (Optional.) Configuring IS-IS timers
   - Specifying the interval for sending IS-IS hello packets
   - Specifying the interval for sending IS-IS CSNP packets
   - Setting the maximum age of LSPs
   - Setting the LSP refresh interval and generation interval
   - Setting LSP sending intervals
   - Setting the SPF calculation interval
5. (Optional.) Configuring IS-IS packet-related features
   - Configuring a DIS priority for an interface
   - Configuring the tag value for an interface
   - Specifying the IS-IS hello multiplier
   - Disabling an interface from sending/receiving IS-IS packets
   - Enabling an interface to send small hello packets
   - Setting LSP lengths
   - Enabling LSP flash flooding
   - Enabling LSP fragment extension
6. (Optional.) Configuring advanced IS-IS features

- Enabling source address check for hello packets on a P2P interface
- Configuring convergence priorities for specific routes
- Setting the LSDB overload bit
- Configuring the ATT bit
- Enabling IS-IS to group ECMP routes

7. (Optional.) Configuring IS-IS logging and SNMP notifications
   - Enabling the logging of neighbor state changes
   - Configuring IS-IS network management
8. (Optional.) Configuring IS-IS fast convergence
   - Enabling ISPF
   - Enabling prefix suppression
   - Configuring IS-IS PIC
9. (Optional.) Enhancing IS-IS network security
   - Configuring neighbor relationship authentication
   - Configuring area authentication
   - Configuring routing domain authentication
10. (Optional.) Enhancing IS-IS network reliability
    - Configuring IS-IS GR
    - Configuring IS-IS NSR
    - Configuring BFD for IS-IS
    - Controlling adjacency establishment and maintenance based on BFD session state
    - Configuring IS-IS FRR

Configuring basic IS-IS

Enabling IPv4 IS-IS

1. Enter system view.

   system-view
2. Enable IS-IS and enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   By default, IS-IS is disabled.
3. Assign a NET.

   network-entity *net*

   By default, NET is not assigned.

CAUTION:

When you execute the **network-entity** command together with the **cost-style** and **is-level** commands for the same IS-IS process, execute the **network-entity** command at last. Incorrect configuration order might cause data loss because the IS-IS process will restart.

4. Return to system view.

   quit
5. Enter interface view.

interface *interface-type interface-number*

6. Enable IS-IS on the interface.

   isis enable [*process-id*]

   By default, IS-IS is disabled.

Enabling IPv6 IS-IS

1. Enter system view.

   system-view

2. Enable an IS-IS process and enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   By default, no IS-IS process is enabled.

3. Configure the NET for the IS-IS process.

   network-entity *net*

   By default, the NET is not configured.

△ **CAUTION:**

When you execute the **network-entity** command together with the **cost-style** and **is-level** commands for the same IS-IS process, execute the **network-entity** command at last. Incorrect configuration order might cause data loss because the IS-IS process will restart.

4. Create the IPv6 address family and enter its view.

   address-family ipv6 [**unicast**]

5. Return to IS-IS view.

   quit

6. Return to system view.

   quit

7. Enter interface view.

   interface *interface-type interface-number*

8. Enable IPv6 for IS-IS on the interface.

   isis ipv6 enable [*process-id*]

   By default, IPv6 is disabled for IS-IS on an interface.

Setting the IS level and circuit level

About this task

Follow these guidelines when you configure the IS level for routers in only one area:

- Set the IS level of all routers to Level-1 or Level-2 rather than different levels because the routers do not need to maintain two identical LSDBs.
- Set the IS level to Level-2 on all routers in an IP network for good scalability.

For an interface of a Level-1 or Level-2 router, the circuit level can only be Level-1 or Level-2. For an interface of a Level-1-2 router, the default circuit level is Level-1-2. If the router only needs to form Level-1 or Level-2 neighbor relationships, set the circuit level for its interfaces to Level-1 or Level-2. This will limit neighbor relationship establishment.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify the IS level.

   is-level { **level-1** | **level-1-2** | **level-2** }

   By default, the IS level is Level-1-2.

4. Return to system view.

   quit

5. Enter interface view.

   interface *interface-type interface-number*

6. Specify the circuit level.

   isis circuit-level [**level-1** | **level-1-2** | **level-2**]

   By default, an interface can establish either the Level-1 or Level-2 adjacency.

Configuring P2P network type for an interface

About this task

Interfaces with different network types operate differently. For example, broadcast interfaces on a network must elect the DIS and flood CSNP packets to synchronize the LSDBs. However, P2P interfaces on a network do not need to elect the DIS, and have a different LSDB synchronization mechanism.

If only two routers exist on a broadcast network, set the network type of attached interfaces to P2P. This avoids DIS election and CSNP flooding, saving network bandwidth and speeding up network convergence.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure P2P network type for an interface.

   isis circuit-type p2p

   By default, the network type of an interface is broadcast.

   Perform this task only for a broadcast network that has up to two attached routers.

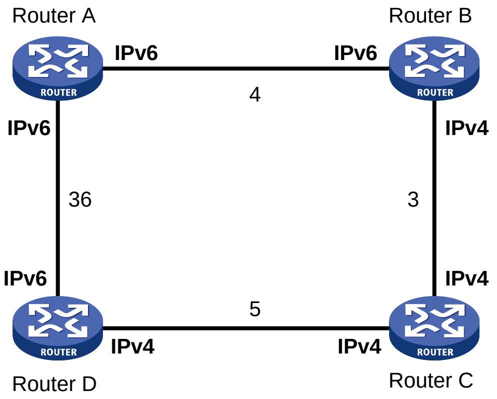
Configuring IPv6 IS-IS MTR

About this task

On a network, IPv4 and IPv6 topologies must be consistent so that both IPv6 IS-IS and IPv4 IS-IS can use the SPF algorithm to perform route calculation. If they are different, routers supporting both IPv4 and IPv6 might send IPv6 packets to routers that do not support IPv6, resulting in packet loss.

To resolve this issue, configure IPv6 IS-IS MTR to perform route calculation separately in IPv4 and IPv6 topologies.

Figure 7 Network diagram



As shown in Figure 7, the numbers refer to the link costs. Router A, Router B, and Router D support both IPv4 and IPv6. Router C supports only IPv4 and cannot forward IPv6 packets.

Enable IPv6 IS-IS MTR on Router A, Router B, Router C, and Router D to make them perform route calculation separately in IPv4 and IPv6 topologies. With this configuration, Router A does not forward IPv6 packets destined to Router D through Router B, avoiding packet loss.

Restrictions and guidelines

As a best practice to avoid route calculation failures, configure this feature when both IPv4 and IPv6 topologies exist in the network.

Procedure

1. Enter system view.

`system-view`

2. Enter IS-IS view.

`isis` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

3. Specify an IS-IS cost style.

`cost-style` { **`compatible`** | **`wide`** | **`wide-compatible`** }

By default, IS-IS only transmits and receives packets using the **`narrow`** cost style.

4. Enter IPv6 address family view.

`address-family ipv6` [**`unicast`**]

5. Enable IPv6 IS-IS MTR.

`multi-topology` [**`compatible`**]

By default, IPv6 IS-IS MTR is disabled.

Configuring IS-IS route control

Configuring IS-IS link cost

About this task

The IS-IS cost of an interface is determined in the following order:

1. IS-IS cost specified in interface view.

2. IS-IS cost specified in system view.

The cost is applied to the interfaces associated with the IS-IS process.

3. Automatically calculated cost.

If the cost style is **wide** or **wide-compatible**, IS-IS automatically calculates the cost using the formula: Interface cost = (Bandwidth reference value / Expected interface bandwidth) × 10, in the range of 1 to 16777214. For other cost styles, Table 3 applies.

Configure the expected bandwidth of an interface with the **bandwidth** command.

Table 3 Automatic cost calculation scheme for cost styles other than wide and wide-compatible

Interface bandwidth	Interface cost
≤ 10 Mbps	60
≤ 100 Mbps	50
≤ 155 Mbps	40
≤ 622 Mbps	30
≤ 2500 Mbps	20
> 2500 Mbps	10

4. If none of the above costs is used, a default cost of 10 applies.

Configuring an IPv4 IS-IS cost for an interface

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. (Optional.) Specify an IS-IS cost style.

   cost-style { **narrow** | **wide** | **wide-compatible** | { **compatible** | **narrow-compatible** } [**relax-spf-limit**] }

   By default, the IS-IS cost type is **narrow**.

4. Return to system view.

   quit

5. Enter interface view.

   interface *interface-type interface-number*

6. Specify a cost for the IS-IS interface.

   isis cost *cost-value* [**level-1** | **level-2**]

   By default, no cost for the interface is specified.

Configuring a global IPv4 IS-IS cost

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify a global IS-IS cost.

   circuit-cost *cost-value* [**level-1** | **level-2**]

   By default, no global cost is specified.

Enabling automatic IPv4 IS-IS cost calculation

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable automatic IS-IS cost calculation.

   auto-cost enable

   By default, automatic IS-IS cost calculation is disabled.

4. (Optional.) Configure a bandwidth reference value for automatic IS-IS cost calculation.

   bandwidth-reference *value*

   The default setting is 100 Mbps.

Configuring an IPv6 IS-IS cost for an interface

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. (Optional.) Specify an IS-IS cost style.

   cost-style { **narrow** | **wide** | **wide-compatible** | { **compatible** | **narrow-compatible** } [**relax-spf-limit**] }

   By default, the IS-IS cost type is **narrow**.

4. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

5. Return to IS-IS view.

   quit

6. Return to system view.

   quit

7. Enter interface view.

   interface *interface-type interface-number*

8. Enable IPv6 for IS-IS on the interface.

   isis ipv6 enable [*process-id*]

   By default, IPv6 is disabled for IS-IS on an interface.

9. Specify an IPv6 cost for the IS-IS interface.

   isis ipv6 cost *cost-value* [**level-1** | **level-2**]

   By default, no IPv6 cost is specified for the interface.

Configuring a global IPv6 IS-IS cost

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Specify a global IPv6 IS-IS cost.

   circuit-cost *cost-value* [**level-1** | **level-2**]

   By default, no global IPv6 cost is specified.

Enabling automatic IPv6 IS-IS cost calculation

1. Enter system view.

system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify an IS-IS cost style.

   cost-style { **wide** | **wide-compatible** }

   By default, the IS-IS cost style is **narrow**.

4. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

5. Enable automatic IPv6 IS-IS cost calculation.

   auto-cost enable

   By default, automatic IPv6 IS-IS cost calculation is disabled.

6. (Optional.) Configure a bandwidth reference value for automatic IPv6 IS-IS cost calculation.

   bandwidth-reference *value*

   By default, the bandwidth reference value is 100 Mbps.

Specifying a preference for IS-IS

About this task

If multiple routing protocols find routes to the same destination, the route found by the routing protocol that has the highest preference is selected as the optimal route.

Perform this task to assign a preference to IS-IS directly or by using a routing policy. For more information about the routing policy, see "Configuring routing policies."

Configuring a preference for IPv4 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Configure a preference for IPv4 IS-IS.

   preference { *preference* | **route-policy** *route-policy-name* } *

   The default setting is 15.

Configuring a preference for IPv6 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Configure a preference for IPv6 IS-IS.

   preference { **route-policy** *route-policy-name* | *preference* } *

   The default setting is 15.

Configuring the maximum number of ECMP routes

About this task

Perform this task to implement load sharing over ECMP routes.

Configuring the maximum number of ECMP routes for IPv4 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Specify the maximum number of ECMP routes.

   maximum load-balancing *number*

   By default, the maximum number of ECMP routes supported by IPv4 IS-IS equals the maximum number of ECMP routes supported by the system.

Configuring the maximum number of ECMP routes for IPv6 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Specify the maximum number of ECMP routes.

   maximum load-balancing *number*

   By default, the maximum number of ECMP routes supported by IPv6 IS-IS equals the maximum number of ECMP routes supported by the system.

Configuring IS-IS route summarization

About this task

Perform this task to summarize specific routes, including IS-IS routes and redistributed routes, into a single route. Route summarization can reduce the routing table size and the LSDB scale.

Route summarization applies only to locally generated LSPs.

Configuring IPv4 IS-IS route summarization

1. Enter system view.

   system-view

2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Configure IPv4 IS-IS route summarization.

   summary *ip-address* { *mask-length* | *mask* } [**avoid-feedback** | **generate_null0_route** | [**level-1** | **level-1-2** | **level-2**] | **tag** *tag*] *

   By default, IPv4 IS-IS route summarization is not configured.

   The cost of the summary route is the lowest one among the costs of the more-specific routes.

Configuring IPv6 IS-IS route summarization

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Configure IPv6 IS-IS route summarization.

   summary *ipv6-prefix prefix-length* [**avoid-feedback** | **generate_null0_route** | [**level-1** | **level-1-2** | **level-2**] | **tag** *tag*] *

   By default, IPv6 IS-IS route summarization is not configured.

Advertising a default route

About this task

IS-IS cannot redistribute a default route to its neighbors. This task enables IS-IS to advertise a default route of 0.0.0.0/0 in an LSP to the same-level neighbors. Upon receiving the default route, the neighbors add it into their routing table.

Advertising an IPv4 IS-IS default route

1. Enter system view.

   system-view

2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Advertise a Level-1 or Level-2 default route.

   default-route-advertise [[**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name*] *

   By default, IPv4 IS-IS does not advertise a Level-1 or Level-2 default route.

Advertising an IPv6 IS-IS default route

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Advertise a Level-1 or Level-2 default route.

   default-route-advertise [**avoid-learning** | [**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name* | **tag** *tag*] *

   By default, IPv6 IS-IS does not advertise a Level-1 or Level-2 default route.

Configuring IS-IS route redistribution

About this task

Perform this task to redistribute routes from other routing protocols into IS-IS. You can specify a cost for redistributed routes and specify the maximum number of redistributed routes.

Restrictions and guidelines

This command redistributes only active routes. To display active routes, use the **`display ip routing-table protocol`** command.

Configuring IPv4 IS-IS route redistribution

1. Enter system view.

   ```
   system-view
   ```

2. Enter IS-IS IPv4 unicast address family view.

   ```
   isis [ process-id ] [ vpn-instance vpn-instance-name ]
   address-family ipv4 [ unicast ]
   ```

3. Redistribute routes from other routing protocols or other IS-IS processes.

   ```
   import-route bgp [ as-number ] [ allow-ibgp ] [ cost cost-value |
   cost-type { external | internal } | [ level-1 | level-1-2 | level-2 ] |
   route-policy route-policy-name | tag tag ] *
   ```

   ```
   import-route { direct | static } [ cost cost-value | cost-type { external
   | internal } | [ level-1 | level-1-2 | level-2 ] | route-policy
   route-policy-name | tag tag ] *
   ```

   ```
   import-route [ allow-direct | cost cost-value | cost-type { external |
   internal } | [ level-1 | level-1-2 | level-2 ] | route-policy
   route-policy-name | tag tag ] *
   ```

   ```
   import-route { isis | ospf | rip } [ process-id | all-processes ]
   [ allow-direct | cost cost-value | cost-type { external | internal } |
   [ level-1 | level-1-2 | level-2 ] | route-policy route-policy-name | tag
   tag ] *
   ```

   By default, IS-IS does not redistribute routes.

4. (Optional.) Configure the maximum number of redistributed Level 1/Level 2 IPv4 routes.

   ```
   import-route limit number
   ```

   By default, IS-IS does not redistribute Level 1/Level 2 IPv4 routes.

Configuring IPv6 IS-IS route redistribution

1. Enter system view.

   ```
   system-view
   ```

2. Enter IS-IS view.

   ```
   isis [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Enter IS-IS IPv6 address family view.

   ```
   address-family ipv6 [ unicast ]
   ```

4. Redistribute routes from other routing protocols or other IS-IS processes.

   ```
   import-route protocol [ as-number | process-id ] [ allow-ibgp ]
   [ allow-direct | cost cost-value | [ level-1 | level-1-2 | level-2 ] |
   route-policy route-policy-name | tag tag ] *
   ```

   By default, IPv6 IS-IS does not redistribute routes.

5. (Optional.) Configure the maximum number of redistributed Level 1/Level 2 IPv6 routes.

import-route limit *number*

By default, IS-IS does not redistribute Level 1/Level 2 IPv6 routes.

Filtering routes calculated from received LSPs

About this task

IS-IS saves LSPs received from neighbors in the LSDB, and uses the SPF algorithm to calculate the shortest path tree with itself as the root. IS-IS installs the calculated routes to the IS-IS routing table and the optimal routes to the IP routing table.

Perform this task to filter calculated routes. Only routes that are not filtered can be added to the IP routing table. The filtered routes retain in the IS-IS routing table and can be advertised to neighbors.

Filtering IPv4 IS-IS routes calculated from received LSPs

1. Enter system view.

   system-view
2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]
3. Filter routes calculated using received LSPs.

   filter-policy { *ipv4-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } **import**

   By default, IPv4 IS-IS route filtering is not configured.

Filtering IPv6 IS-IS routes calculated from received LSPs

1. Enter system view.

   system-view
2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]
3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]
4. Filter routes calculated using received LSPs.

   filter-policy { *ipv6-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } **import**

   By default, IPv6 IS-IS route filtering is not configured.

Filtering redistributed routes

About this task

IS-IS can redistribute routes from other routing protocols or other IS-IS processes, add them to the IS-IS routing table, and advertise them in LSPs.

Perform this task to filter redistributed routes. Only routes that are not filtered can be added to the IS-IS routing table and advertised to neighbors.

Restrictions and guidelines

Use this command together with the **import-route** command.

Filtering redistributed IPv4 IS-IS routes

1. Enter system view.

```
system-view
```

2. Enter IS-IS IPv4 unicast address family view.

   ```
   isis [ process-id ] [ vpn-instance vpn-instance-name ]
   address-family ipv4 [ unicast ]
   ```

3. Filter routes redistributed from other routing protocols or IS-IS processes.

   ```
   filter-policy { ipv4-acl-number | prefix-list prefix-list-name |
   route-policy route-policy-name } export [ bgp | direct | { isis | ospf |
   rip } process-id | static ]
   ```

   By default, IPv4 IS-IS route filtering is not configured.

Filtering redistributed IPv6 IS-IS routes

1. Enter system view.

   ```
   system-view
   ```

2. Enter IS-IS view.

   ```
   isis [ process-id ] [ vpn-instance vpn-instance-name ]
   ```

3. Enter IS-IS IPv6 address family view.

   ```
   address-family ipv6 [ unicast ]
   ```

4. Filter routes redistributed from other routing protocols or IS-IS processes.

   ```
   filter-policy { ipv6-acl-number | prefix-list prefix-list-name |
   route-policy route-policy-name } export [ bgp4+ | direct | { isisv6 |
   ospfv3 | ripng } process-id | static ]
   ```

   By default, IPv6 IS-IS route filtering is not configured.

Configuring IS-IS route leaking

About this task

Perform this task to control route advertisement (route leaking) between Level-1 and Level-2.

You can configure IS-IS to advertise routes from Level-2 to Level-1, and to not advertise routes from Level-1 to Level-2.

Configuring IPv4 IS-IS route leaking

1. Enter system view.

   ```
   system-view
   ```

2. Enter IS-IS IPv4 unicast address family view.

   ```
   isis [ process-id ] [ vpn-instance vpn-instance-name ]
   address-family ipv4 [ unicast ]
   ```

3. Configure route leaking from Level-1 to Level-2.

   ```
   import-route isis level-1 into level-2 [ filter-policy
   { ipv4-acl-number | prefix-list prefix-list-name | route-policy
   route-policy-name } | tag tag ] *
   ```

   By default, IS-IS advertises routes from Level-1 to Level-2.

4. Configure route leaking from Level-2 to Level-1.

   ```
   import-route isis level-2 into level-1 [ filter-policy
   { ipv4-acl-number | prefix-list prefix-list-name | route-policy
   route-policy-name } | tag tag ] *
   ```

   By default, IS-IS does not advertise routes from Level-2 to Level-1.

Configuring IPv6 IS-IS route leaking

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Configure route leaking from Level-2 to Level-1.

   import-route isisv6 level-2 into level-1 [**filter-policy** { *ipv6-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } | **tag** *tag*] *

   By default, IS-IS does not advertise routes from Level-2 to Level-1.

5. Configure route leaking from Level-1 to Level-2.

   import-route isisv6 level-1 into level-2 [**filter-policy** { *ipv6-acl-number* | **prefix-list** *prefix-list-name* | **route-policy** *route-policy-name* } | **tag** *tag*] *

   By default, IS-IS advertises routes from Level-1 to Level-2.

Advertising IS-IS link state information to BGP

About this task

After the device advertises IS-IS link state information to BGP, BGP can advertise the information for intended applications. For more information about BGP LS, see "Configuring BGP."

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Advertise IS-IS link state information to BGP.

   distribute bgp-ls [**instance-id** *id*] [**level-1** | **level-2**]

   By default, the device does not advertise IS-IS link state information to BGP.

Configuring IS-IS timers

Specifying the interval for sending IS-IS hello packets

About this task

If a neighbor does not receive any hello packets from the router within the advertised hold time, it considers the router down and recalculates the routes. The hold time is the hello multiplier multiplied by the hello interval.

Restrictions and guidelines

The interval between hello packets sent by the DIS is 1/3 the hello interval set with the **isis timer hello** command.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the interval for sending hello packets.

   isis timer hello *seconds* [**level-1** | **level-2**]

   The default setting is 10 seconds.

Specifying the interval for sending IS-IS CSNP packets

About this task

On a broadcast network, perform this task on the DIS that uses CSNP packets to synchronize LSDBs.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the interval for sending CSNP packets on the DIS of a broadcast network.

   isis timer csnp *seconds* [**level-1** | **level-2**]

   The default setting is 10 seconds.

Setting the maximum age of LSPs

About this task

Each LSP has an age that decreases in the LSDB. Any LSP with an age of 0 is deleted from the LSDB. You can adjust the age value based on the scale of a network.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the maximum LSP age.

   timer lsp-max-age *seconds*

   The default setting is 1200 seconds.

Setting the LSP refresh interval and generation interval

About this task

Each router needs to refresh its LSPs at a configurable interval and send them to other routers to prevent valid routes from aging out. A smaller refresh interval speeds up network convergence but consumes more bandwidth.

When network topology changes such as neighbor state, interface metric, system ID, or area ID changes occur, the router generates an LSP after a configurable interval. If such a change occurs frequently, excessive LSPs are generated, consuming a large amount of router resources and bandwidth. To solve the problem, you can adjust the LSP generation interval.

Restrictions and guidelines

Follow these restrictions and guidelines when you configure the **timer lsp-generation** command:

- If you specify only the *maximum-interval* argument, the LSP generation interval is *maximum-interval*.
- If you do not specify the *incremental-interval* argument, the LSP generation interval is in the range of *minimum-interval* to *maximum-interval*.
- If you specify the *incremental-interval* argument, the LSP generation interval is as follows:
    - When network changes are not frequent, the *minimum-interval* is adopted.
    - If network changes are frequent, the LSP generation interval increases by *incremental-interval* × 2^{n-2} (n is the number of calculation times) each time a generation occurs until the *maximum-interval* is reached.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the LSP refresh interval.

   timer lsp-refresh *seconds*

   By default, the LSP refresh interval is 900 seconds.

4. Set the LSP generation interval.

   timer lsp-generation *maximum-interval* [*minimum-interval* [*incremental-interval*]] [**level-1** | **level-2**]

   By default:

    - The maximum interval is 5 seconds.
    - The minimum interval is 50 milliseconds.
    - The incremental interval is 200 milliseconds.

Setting LSP sending intervals

About this task

If a change occurs in the LSDB, IS-IS advertises the changed LSP to neighbors. You can specify the minimum interval for sending these LSPs to control the amount of LSPs on the network.

On a P2P link, IS-IS requires an advertised LSP be acknowledged. If no acknowledgment is received within a configurable interval, IS-IS will retransmit the LSP.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the minimum interval for sending LSPs and the maximum LSP number that can be sent at a time.

   isis timer lsp *time* [**count** *count*]

   By default, the minimum interval is 33 milliseconds, and the maximum LSP number that can be sent at a time is 5.

4. Specify the LSP retransmission interval on a P2P link.

   isis timer retransmit *seconds*

   By default, the LSP retransmission interval on a P2P link is 5 seconds.

Setting the SPF calculation interval

About this task

Based on the LSDB, an IS-IS router uses the SPF algorithm to calculate the shortest path tree with itself being the root, and uses the shortest path tree to determine the next hop to a destination network. By adjusting the SPF calculation interval, you can prevent bandwidth and router resources from being over consumed due to frequent topology changes.

When network changes are not frequent, the *minimum-interval* is adopted. If network changes become frequent, the SPF calculation interval increases by *incremental-interval* × 2^{n-2} (n is the number of calculation times) each time a calculation occurs until the *maximum-interval* is reached.

Setting the IPv4 SPF calculation interval

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the SPF calculation interval.

   timer spf *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   By default:

   - The maximum interval is 5 seconds.
   - The minimum interval is 50 milliseconds.
   - The incremental interval is 200 milliseconds.

Setting the IPv6 SPF calculation interval

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Set the SPF calculation interval.

   timer spf *maximum-interval* [*minimum-interval* [*incremental-interval*]]

   By default:

   - The maximum interval is 5 seconds.
   - The minimum interval is 50 milliseconds.

- The incremental interval is 200 milliseconds.

Configuring IS-IS packet-related features

Configuring a DIS priority for an interface

About this task

On a broadcast network, IS-IS must elect a router as the DIS at a routing level. You can specify a DIS priority at a level for an interface. The greater the interface's priority, the more likely it becomes the DIS. If multiple routers in the broadcast network have the same highest DIS priority, the router with the highest MAC address becomes the DIS.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure a DIS priority for the interface.

   `isis dis-priority` *`priority`* [**`level-1`** | **`level-2`**]

   The default setting is 64.

Configuring the tag value for an interface

About this task

Perform this task when the link cost style is **wide**, **wide-compatible**, or **compatible**.

When IS-IS advertises a prefix with a tag value, IS-IS adds the tag to the IP reachability information TLV of the prefix.

Configuring the IPv4 IS-IS tag value for an interface

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the IPv4 IS-IS tag value for the interface.

   `isis tag` *`tag`*

   By default, the IPv4 IS-IS tag value of the interface is not configured.

Configuring the IPv6 IS-IS tag value for an interface

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the IPv6 IS-IS tag value for the interface.

   `isis ipv6 tag` *`tag`*

   By default, the IPv6 IS-IS tag value of the interface is not configured.

   When IS-IS advertises an IPv6 prefix with a tag value, it adds the tag to the IPv6 reachability information TLV, regardless of the link cost style.

Specifying the IS-IS hello multiplier

About this task

The hello multiplier is the number of hello packets a neighbor must miss before it declares that the router is down.

If a neighbor receives no hello packets from the router within the advertised hold time, it considers the router down and recalculates the routes. The hold time is the hello multiplier multiplied by the hello interval.

On a broadcast link, Level-1 and Level-2 hello packets are advertised separately. You must set a hello multiplier for each level.

On a P2P link, Level-1 and Level-2 hello packets are advertised in P2P hello packets. You do not need to specify Level-1 or Level-2.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify the hello multiplier.

   `isis timer holding-multiplier` *`value`* [**`level-1`** | **`level-2`**]

   The default setting is 3.

Disabling an interface from sending/receiving IS-IS packets

About this task

After being disabled from sending and receiving hello packets, an interface cannot form any neighbor relationship, but can advertise directly connected networks in LSPs through other interfaces. This can save bandwidth and CPU resources, and ensures that other routers know networks directly connected to the interface.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Disable the interface from sending and receiving IS-IS packets.

   `isis silent`

   By default, the interface can send and receive IS-IS packets.

Enabling an interface to send small hello packets

About this task

IS-IS messages cannot be fragmented at the IP layer because they are directly encapsulated in frames. Any two IS-IS neighboring routers must negotiate a common MTU. To avoid sending big hellos to save bandwidth, enable the interface to send small hello packets without CLVs.

Procedure

1. Enter system view.

system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the interface to send small hello packets without CLVs.

   isis small-hello

   By default, the interface sends standard hello packets.

Setting LSP lengths

About this task

IS-IS messages cannot be fragmented at the IP layer because they are directly encapsulated in frames. IS-IS routers in an area must send LSPs smaller than the smallest interface MTU in the area.

If the IS-IS routers have different interface MTUs, configure the maximum size of generated LSP packets to be smaller than the smallest interface MTU in the area. Without the configuration, the routers must dynamically adjust the LSP packet size to fit the smallest interface MTU, which takes time and affects other services.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify the maximum length of generated Level-1 LSPs or Level-2 LSPs.

   lsp-length originate *size* [**level-1** | **level-2**]

   By default, the maximum length of generated Level-1 LSPs or Level-2 LSPs is 1497 bytes.

4. Specify the maximum length of received LSPs.

   lsp-length receive *size*

   By default, the maximum length of received LSPs is 1497 bytes.

Enabling LSP flash flooding

About this task

Changed LSPs can trigger SPF recalculation. To advertise the changed LSPs before the router recalculates routes for faster network convergence, enable LSP flash flooding.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable LSP flash flooding.

   flash-flood [**flood-count** *flooding-count* | **max-timer-interval** *flooding-interval* | [**level-1** | **level-2**]] *

   By default, LSP flash flooding is disabled.

Enabling LSP fragment extension

1. Enter system view.

   `system-view`

2. Enter IS-IS view.

   `isis` [*`process-id`*] [**`vpn-instance`** *`vpn-instance-name`*]

3. Enable LSP fragment extension.

   `lsp-fragments-extend` [**`level-1`** | **`level-1-2`** | **`level-2`**]

   By default, LSP fragment extension is disabled.

   The MTUs of all interfaces running the IS-IS process must not be less than 512. Otherwise, LSP fragment extension does not take effect.

4. Configure a virtual system ID.

   `virtual-system` *`virtual-system-id`*

   By default, no virtual system ID is configured.

   Configure a minimum of one virtual system to generate extended LSP fragments.

Configuring advanced IS-IS features

Enabling source address check for hello packets on a P2P interface

About this task

An IS-IS P2P interface can have a peer on a different network. Perform this task to configure an IS-IS P2P interface to establish neighbor relationship only with a peer on the same network.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable source address check for hello packets on a P2P interface.

   `isis peer-ip-check`

   By default, an IS-IS P2P interface can have a peer on a different network.

Configuring convergence priorities for specific routes

About this task

A topology change causes IS-IS routing convergence. To improve convergence speed, you can assign convergence priorities to IS-IS routes. Convergence priority levels are critical, high, medium, and low. The higher the convergence priority, the faster the convergence speed.

By default, IS-IS host routes have medium convergence priority, and other IS-IS routes have low convergence priority.

Configuring convergence priorities for specific IPv4 IS-IS routes

1. Enter system view.

   `system-view`

2. Enter IS-IS IPv4 unicast address family view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Assign convergence priorities to specific IPv4 IS-IS routes.
   - Assign a convergence priority to IPv4 IS-IS routes matching the specified prefix list.

     prefix-priority { **critical** | **high** | **medium** } { **prefix-list** *prefix-list-name* | **tag** *tag-value* }
   - Assign a convergence priority to IPv4 IS-IS routes by using a route policy.

     prefix-priority route-policy *route-policy-name*

   By default, IPv4 IS-IS routes, except IS-IS host routes, have the low convergence priority.

Configuring convergence priorities for specific IPv6 IS-IS routes

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Assign convergence priorities to specific IPv6 IS-IS routes.

   prefix-priority { **critical** | **high** | **medium** } { **prefix-list** *prefix-list-name* | **tag** *tag-value* }

   prefix-priority route-policy *route-policy-name*

   By default, IPv6 IS-IS routes, except IS-IS host routes, have the low convergence priority.

Setting the LSDB overload bit

About this task

By setting the overload bit in sent LSPs, a router informs other routers of failures that make it unable to select routes and forward packets.

When an IS-IS router cannot record the complete LSDB, for example, because of memory insufficiency, it will calculate wrong routes. To make troubleshooting easier, temporarily isolate the router from the IS-IS network by setting the overload bit.

Setting the LSDB overload bit for IPv4 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the overload bit.

   set-overload [**on-startup** [[**start-from-nbr** *system-id* [*timeout1* [*nbr-timeout*]]] | *timeout2* | **wait-for-bgp** [*timeout3*]]] [**allow** { **external** | **interlevel** } *]

   By default, the overload bit is not set.

Setting the LSDB overload bit for IPv6 IS-IS

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Set the overload bit.

   set-overload [**on-startup** [[**start-from-nbr** *system-id* [*timeout1* [*nbr-timeout*]]] | *timeout2* | **wait-for-bgp4+** [*timeout3*]]] [**allow** { **external** | **interlevel** } *]

   By default, the overload bit is not set.

Configuring the ATT bit

About this task

The ATT bit is used to identify the connection status between a Level-1 area and other areas. By default, a Level-1-2 router sets the ATT bit for Level-1 LSPs as follows:

- The Level-1-2 router sets the ATT bit in Level-1 LSPs to inform the Level-1 routers that it can reach other areas. After a Level-1 router receives a Level-1 LSP with the ATT bit set, it generates a default route destined for the Level-1-2 router.
- The Level-1-2 router does not set the ATT bit in Level-1 LSPs if it can reach only one area.

To edit the default ATT bit setting rule for a Level-1-2 router, perform the following tasks as needed:

- To enable ATT bit setting for all Level-1 LSPs, execute the **set-att always** command on the Level-1-2 router.
- To disable a Level-1 router from generating a default route upon receiving an ATT-bit-set Level-1 LSP from the Level-1-2 router, you can perform one of the following tasks:
  - Execute the **ignore-att** command on the Level-1 router.
  - Execute the **set-att never** command on the Level-1-2 router.

Configuring IS-IS not to calculate the default route through the ATT bit

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Configure IS-IS not to calculate the default route through the ATT bit.

   ignore-att

   By default, IS-IS uses the ATT bit to calculate the default route.

Setting the ATT bit of IPv4 Level-1 LSPs

1. Enter system view.

   system-view

2. Enter IS-IS view

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Set the ATT bit of IPv4 Level-1 LSPs.

   set-att { **always** | **never** }

   By default, the Level-1-2 router sets the ATT bit for IPv4 Level-1 LSPs in accordance with the default ATT bit setting rule.

Setting the ATT bit of IPv6 Level-1 LSPs

1. Enter system view.

   system-view

2. Enter IS-IS view

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Set the ATT bit of IPv6 Level-1 LSPs.

   set-att { **always** | **never** }

   By default, the Level-1-2 router sets the ATT bit for IPv6 Level-1 LSPs in accordance with the default ATT bit setting rule.

Configuring system ID to host name mappings

About this task

A 6-byte system ID in hexadecimal notation uniquely identifies a router or host in an IS-IS network. To make a system ID easy to read, the system allows you to use host names to identify devices. It also provides mappings between system IDs and host names.

The mappings can be configured manually or dynamically.

- **Static system ID to host name mapping**—You must manually configure a mapping for each router in the network. When a new router is added to the network or a mapping must be modified, you must configure all routers manually.
- **Dynamic system ID to host name mapping**—You only need to configure a host name for each router in the network. Each router advertises the host name in a dynamic host name CLV to other routers so all routers in the network can have all mappings. To help check the origin of LSPs in the LSDB, you can configure a name for the DIS in a broadcast network.

Restrictions and guidelines

Follow these guidelines when you configure the mappings:

- To view host names rather than system IDs by using the **display isis lsdb** command, you must enable dynamic system ID to host name mapping.
- If you configure both dynamic mapping and static mapping on a router, the host name specified for dynamic mapping applies.

Configuring a static system ID to host name mapping

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Configure a system ID to host name mapping for a remote IS.

   is-name map *sys-id map-sys-name*

   By default, no system ID to host name mapping is configured for a remote IS.

   A system ID can correspond to only one host name.

Configuring dynamic system ID to host name mapping

1. Enter system view.

   system-view

2. Enter IS-IS view.

isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify a host name for the IS and enable dynamic system ID to host name mapping.

   is-name *sys-name*

   By default, dynamic system ID to host name mapping is disabled and no host name is specified for the router.

4. Return to system view.

   quit

5. Enter interface view.

   interface *interface-type interface-number*

6. Configure a DIS name.

   isis dis-name *symbolic-name*

   By default, no DIS name is configured.

   This command takes effect only on a router enabled with dynamic system ID to host name mapping.

   This command is not available on P2P interfaces.

Enabling IS-IS to group ECMP routes

About this task

Perform this task to enable IS-IS to group ECMP routes by prefix to speed up route convergence.

This feature is applicable to a network when the network has a large number of ECMP routes and different route prefixes in the network have the same next hops. For example, IS-IS learns 10000 route prefixes and all route prefixes have the same 16 next hops (1.1.1.1 to 1.1.1.16). Without this feature, IS-IS has to send all ECMP routes of every route prefix (10000 × 16 routes) to the route management module. After you enable this feature, IS-IS groups the ECMP routes by prefix and sends the route groups (10000 route groups) to the route management module.

Restrictions and guidelines

If the output interfaces to the next hops of ECMP routes are TE tunnel interfaces, IS-IS groups the ECMP routes regardless of whether you enable this feature or not.

Enabling IPv4 IS-IS to group ECMP routes

1. Enter system view.

   system-view

2. Execute the following commands in sequence to enter IS-IS IPv4 unicast address family view:

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

   address-family ipv4 [**unicast**]

3. Enable IPv4 IS-IS to group ECMP routes.

   ecmp-group enable

   By default, IPv4 IS-IS does not group ECMP routes.

Enabling IPv6 IS-IS to group ECMP routes

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Enable IPv6 IS-IS to group ECMP routes.

   ecmp-group enable

   By default, IPv6 IS-IS does not group ECMP routes.

Configuring IS-IS logging and SNMP notifications

Enabling the logging of neighbor state changes

About this task

With this feature enabled, the router delivers logs about neighbor state changes to its information center. The information center processes the logs according to user-defined output rules (whether to output logs and where to output). For more information about the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable the logging of neighbor state changes.

   log-peer-change

   By default, the logging of neighbor state changes is enabled.

Configuring IS-IS network management

About this task

This task includes the following configurations:

- Bind an IS-IS process to MIB so that you can use network management software to manage the specified IS-IS process.
- Enable IS-IS notifications to report important events.

To report critical IS-IS events to an NMS, enable SNMP notifications for IS-IS. For SNMP notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see the network management and monitoring configuration guide for the device.

Procedure

1. Enter system view.

   system-view

2. Bind MIB to an IS-IS process.

   isis mib-binding *process-id*

   By default, MIB is bound to the IS-IS process with the smallest process ID.

3. Enable IS-IS notification sending.

   snmp-agent trap enable isis [**adjacency-state-change** | **area-mismatch** | **authentication** | **authentication-type** | **buffsize-mismatch** | **id-length-mismatch** | **lsdboverload-state-change** | **lsp-corrupt** | **lsp-parse-error** | **lsp-size-exceeded** | **manual-address-drop** | **max-seq-exceeded** | **maxarea-mismatch** | **own-lsp-purge** |

protocol-support | **rejected-adjacency** | **skip-sequence-number** | **version-skew**] *

By default, IS-IS notification sending is enabled.

4. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

5. Configure the context name for the SNMP object for managing IS-IS.

   snmp context-name *context-name*

   By default, no context name is set for the SNMP object for managing IS-IS.

Configuring IS-IS fast convergence

Enabling ISPF

About this task

When the network topology changes, Incremental Shortest Path First (ISPF) computes only the affected part of the SPT, instead of the entire SPT.

Enabling IPv4 IS-IS ISPF

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable IPv4 IS-IS ISPF.

   ispf enable

   By default, IPv4 IS-IS ISPF is enabled.

Enabling IPv6 IS-IS ISPF

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Enable IPv6 IS-IS ISPF.

   ispf enable

   By default, IPv6 IS-IS ISPF is enabled.

Enabling prefix suppression

About this task

Perform this task to disable an interface from advertising its prefix in LSPs. This enhances network security by preventing IP routing to the interval nodes and speeds up network convergence.

Enabling IPv4 IS-IS prefix suppression

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IPv4 IS-IS prefix suppression on the interface.

   isis prefix-suppression

   By default, IPv4 IS-IS prefix suppression is disabled on the interface.

   This command is also applicable to the secondary IP address of the interface.

Enabling IPv6 IS-IS prefix suppression

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IPv6 IS-IS prefix suppression on the interface.

   isis ipv6 prefix-suppression

   By default, IPv6 IS-IS prefix suppression is disabled on the interface.

Configuring IS-IS PIC

About this task

Prefix Independent Convergence (PIC) enables the device to speed up network convergence by ignoring the number of prefixes.

Restrictions and guidelines for IS-IS PIC

Follow these restrictions and guidelines when you configure IS-IS PIC:

- When both IS-IS PIC and IS-IS FRR are configured, IS-IS FRR takes effect.
- IS-IS PIC applies only to LSPs sent by neighbors.

Enabling IS-IS PIC

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable PIC for IS-IS.

   pic [**additional-path-always**]

   By default, IS-IS PIC is disabled.

Enabling BFD control packet mode for IS-IS PIC

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for IS-IS PIC.

   isis primary-path-detect bfd ctrl

   By default, BFD control packet mode is disabled for IS-IS PIC.

   To use BFD (control packet mode) to detect primary link failures, you must enable BFD control packet mode on both ends of the link.

Enabling BFD echo packet mode for IS-IS PIC

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interface's IP address.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD echo packet mode for IS-IS PIC.

   isis primary-path-detect bfd echo

   By default, BFD echo packet mode is disabled for IS-IS PIC.

   To use BFD (echo packet mode) to detect primary link failures, you only need to enable BFD echo packet mode on one end of the link.

Enhancing IS-IS network security

To enhance the security of an IS-IS network, you can configure IS-IS authentication. IS-IS authentication involves neighbor relationship authentication, area authentication, and routing domain authentication.

Configuring neighbor relationship authentication

About this task

With neighbor relationship authentication configured, an interface adds the key in the specified mode into hello packets to the peer and checks the key in the received hello packets. If the authentication succeeds, it forms the neighbor relationship with the peer.

The authentication mode and key at both ends must be identical.

To prevent packet exchange failure in case of an authentication key change, configure the interface not to check the authentication information in the received packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Specify the authentication mode and key.

   isis authentication-mode { { **gca** *key-id* { **hmac-sha-1** | **hmac-sha-224** | **hmac-sha-256** | **hmac-sha-384** | **hmac-sha-512** } [**nonstandard**] | **md5** | **simple** } { **cipher** | **plain** } *string* | **keychain** *keychain-name* } [**level-1** | **level-2**] [**ip** | **osi**]

   By default, the authentication mode and key are not configured.

4. (Optional.) Configure the interface not to check the authentication information in the received hello packets.

isis authentication send-only [**level-1** | **level-2**]

When the authentication mode and key are configured, the interface checks the authentication information in the received packets by default.

Configuring area authentication

About this task

Area authentication prevents the router from installing routing information from untrusted routers into the Level-1 LSDB. The router encapsulates the authentication key in the specified mode in Level-1 packets (LSP, CSNP, and PSNP). It also checks the key in received Level-1 packets.

Routers in a common area must have the same authentication mode and key.

To prevent packet exchange failure in case of an authentication key change, configure IS-IS not to check the authentication information in the received packets.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify the area authentication mode and key.

   area-authentication-mode { { **gca** *key-id* { **hmac-sha-1** | **hmac-sha-224** | **hmac-sha-256** | **hmac-sha-384** | **hmac-sha-512** } [**nonstandard**] | **md5** | **simple** } { **cipher** | **plain** } *string* | **keychain** *keychain-name* } [**ip** | **osi**]

   By default, the area authentication mode and key are not configured.

4. (Optional.) Configure the interface not to check the authentication information in the received Level-1 packets, including LSPs, CSNPs, and PSNPs.

   area-authentication send-only

   When the authentication mode and key are configured, the interface checks the authentication information in the received packets by default.

Configuring routing domain authentication

About this task

Routing domain authentication prevents untrusted routing information from entering into a routing domain. A router with the authentication configured encapsulates the key in the specified mode into Level-2 packets (LSP, CSNP, and PSNP) and check the key in received Level-2 packets.

All the routers in the backbone must have the same authentication mode and key.

To prevent packet exchange failure in case of an authentication key change, configure IS-IS not to check the authentication information in the received packets.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Specify the routing domain authentication mode and key.

domain-authentication-mode { { **gca** *key-id* { **hmac-sha-1** | **hmac-sha-224** | **hmac-sha-256** | **hmac-sha-384** | **hmac-sha-512** } [**nonstandard**] | **md5** | **simple** } { **cipher** | **plain** } *string* | **keychain** *keychain-name* } [**ip** | **osi**]

By default, the routing domain authentication mode and key are not configured.

4. (Optional.) Configure the interface not to check the authentication information in the received Level-2 packets, including LSPs, CSNPs, and PSNPs.

domain-authentication send-only

When the authentication mode and key are configured, the interface checks the authentication information in the received packets by default.

Configuring IS-IS GR

About this task

GR ensures forwarding continuity when a routing protocol restarts or an active/standby switchover occurs.

Two routers are required to complete a GR process. The following are router roles in a GR process.

- **GR restarter**—Graceful restarting router. It must have GR capability.
- **GR helper**—A neighbor of the GR restarter. It assists the GR restarter to complete the GR process. By default, the device acts as the GR helper.

Configure IS-IS GR on the GR restarter.

GR restarter uses the following timers:

- **T1 timer**—Specifies the times that GR restarter can send a Restart TLV with the RR bit set. When rebooted, the GR restarter sends a Restart TLV with the RR bit set to its neighbor. If the GR restarter receives a Restart TLV with the RA set from its neighbor before the T1 timer expires, the GR process starts. Otherwise, the GR process fails.
- **T2 timer**—Specifies the LSDB synchronization interval. Each LSDB has a T2 timer. The Level-1-2 router has a Level-1 timer and a Level-2 timer. If the LSDBs have not synchronized before the two timers expire, the GR process fails.
- **T3 timer**—Specifies the GR interval. The GR interval is set as the holdtime in hello PDUs. Within the interval, the neighbors maintain their adjacency with the GR restarter. If the GR process has not completed within the holdtime, the neighbors tear down the neighbor relationship and the GR process fails.

Restrictions and guidelines

IS-IS GR and IS-IS NSR are mutually exclusive. Do not configure them at the same time.

The product of the T1 timer and the number of times that the T1 timer can expire must be smaller than the T2 timer.

The T2 timer must be smaller than the T3 timer.

Procedure

1. Enter system view.

system-view

2. Enable IS-IS and enter IS-IS view.

isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable IS-IS GR.

graceful-restart

By default, the GR capability for IS-IS is disabled.

4. (Optional.) Suppress the SA bit during restart.

graceful-restart suppress-sa

By default, the SA bit is not suppressed.

By enabling the GR restarter to suppress the Suppress-Advertisement (SA) bit in the hello PDUs, the neighbors will still advertise their adjacency with the GR restarter.

5. (Optional.) Configure the T1 timer.

   graceful-restart t1 *seconds* **count** *count*

   By default, the T1 timer is 3 seconds and can expire 10 times.

6. (Optional.) Configure the T2 timer.

   graceful-restart t2 *seconds*

   By default, the T2 timer is 60 seconds.

7. (Optional.) Configure the T3 timer.

   graceful-restart t3 *seconds*

   By default, the T2 timer is 300 seconds.

Configuring IS-IS NSR

About this task

After an active/standby switchover, the GR restarter obtains routing information from its neighbors, and the IS-IS process must learn all the routes. If the network topology changes during the switchover, removed routes cannot be updated to the device, which can result in blackhole routes.

NSR solves the problem by backing up IS-IS link state information from the active process to the standby process. After an active/standby switchover, NSR can complete link state recovery and route regeneration without requiring the cooperation of other devices.

Restrictions and guidelines

IS-IS NSR and IS-IS GR are mutually exclusive. Do not configure them at the same time.

Procedure

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enable IS-IS NSR.

   non-stop-routing

   By default, IS-IS NSR is disabled.

   IS-IS NSR takes effect on a per-process basis. As a best practice, enable NSR for each IS-IS process.

Configuring BFD for IS-IS

About this task

BFD provides a single mechanism to quickly detect and monitor the connectivity of links between IS-IS neighbors, reducing network convergence time.For more information about BFD, see BFD configuration in *Network Management and Monitoring Configuration Guide*.

Configuring BFD for IPv4 IS-IS

1. Enter system view.

```
system-view
```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable BFD on an IPv4 IS-IS interface.

   ```
   isis bfd enable
   ```

   By default, an IPv4 IS-IS interface is not enabled with BFD.

Configuring BFD for IPv6 IS-IS

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable BFD on an IPv6 IS-IS interface.

   ```
   isis ipv6 bfd enable
   ```

   By default, an IPv6 IS-IS interface is not enabled with BFD.

Controlling adjacency establishment and maintenance based on BFD session state

About this task

When BFD detects a Layer 3 forwarding failure between two routers, the BFD session goes down, which causes the IS-IS adjacency to go down. If Layer 2 forwarding is still available, the routers can exchange IS-IS packets and re-establish the adjacency, which might cause traffic loss.

To avoid the issue, enable this feature on the BFD-enabled interfaces of the local and remote routers, enabling the interfaces to carry BFD-enabled TLVs in hello packets. After the BFD session goes down, the routers do not establish an adjacency if the exchanged BFD-enabled TLVs are identical.

If two IS-IS routers establish both IPv4 and IPv6 adjacency relationships, the following rules apply:

- If route calculation is not performed separately for the IPv4 and IPv6 topologies, the IPv4 or IPv6 adjacency relationship is up only when both the IPv4 and IPv6 BFD sessions are up.
- If route calculation is performed separately for the IPv4 and IPv6 topologies, the IPv4 or IPv6 adjacency relationship is up when the corresponding IPv4 or IPv6 BFD session is up.

If two IS-IS routers establish only an IPv4 or IPv6 adjacency relationship, the adjacency relationship is up when the corresponding IPv4 or IPv6 BFD session is up.

For more information about BFD, see BFD configuration in *Network Management and Monitoring Configuration Guide*.

Enabling IPv4 IS-IS adjacency establishment and maintenance control based on BFD session state

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable IPv4 IS-IS BFD on the interface.

   ```
   isis bfd enable
   ```

   By default, IPv4 IS-IS BFD is disabled on an interface.

4. Enable adjacency establishment and maintenance control based on BFD session state.

```
isis bfd session-restrict-adj
```

By default, adjacency establishment and maintenance control based on BFD session state is disabled.

Enabling IPv6 IS-IS adjacency establishment and maintenance control based on BFD session state

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable IPv6 IS-IS BFD on the interface.

   ```
   isis ipv6 bfd enable
   ```

   By default, IPv6 IS-IS BFD is disabled on an interface.

4. Enable adjacency establishment and maintenance control based on BFD session state.

   ```
   isis ipv6 bfd session-restrict-adj
   ```

   By default, adjacency establishment and maintenance control based on BFD session state is disabled.

Configuring IS-IS FRR

About IS-IS FRR

IS-IS Fast Reroute (FRR) calculates a backup path based on the LSDB and saves the backup path information to the FIB. When the primary path fails, the system immdiately switches traffic to the backup path to prevent traffic loss and reduce the route convergence time.

IS-IS supports Loop Free Alternate (LFA) FRR and remote LFA FRR.

The following IS-IS FRR traffic protection types are available:

- **Link protection**—Protects traffic that traverses a specific link.
- **Node protection**—Protects traffic that traverses a specific node.

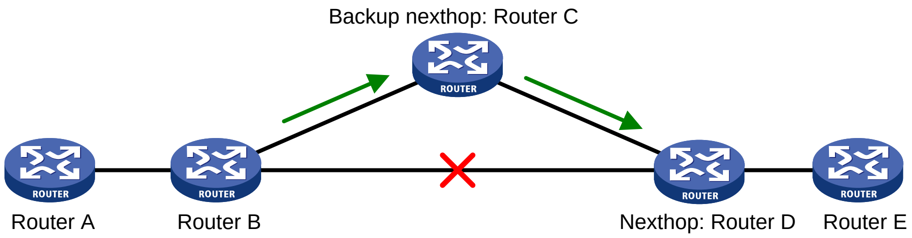
Node protection takes precedence over link protection.

Configuring IS-IS LFA FRR

About this task

A link or router failure on a path can cause packet loss. IS-IS FRR enables fast rerouting to minimize the failover time.

Figure 8 Network diagram for IS-IS FRR



In Figure 8, after you enable FRR on Router B, IS-IS automatically calculates or designates a backup next hop when a link failure is detected. In this way, packets are directed to the backup next hop to reduce traffic recovery time. Meanwhile, IS-IS calculates the shortest path based on the new network topology, and forwards packets over the path after network convergence.

You can assign a backup next hop for IS-IS FRR through the following ways:

- Enable IS-IS FRR to calculate a backup next hop through Loop Free Alternate (LFA) calculation.
- Designate a backup next hop with a routing policy for routes matching specific criteria.

Restrictions and guidelines

The LFA calculation of FRR and that of TE are mutually exclusive.

Configuring IPv4 IS-IS FRR to calculate a backup next hop through LFA calculation

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. (Optional.) Disable LFA calculation on the interface.
   isis fast-reroute lfa-backup exclude [**level-1** | **level-2**]
   By default, the interface participates in LFA calculation, and can be elected as a backup interface.
4. Return to system view.
   quit
5. Enter IS-IS IPv4 unicast address family view.
   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]
   address-family ipv4 [**unicast**]
6. Enable IS-IS FRR to calculate a backup next hop through LFA calculation.
   fast-reroute lfa [**level-1** | **level-2**]
   By default, IS-IS FRR is disabled.

Configuring IPv4 IS-IS FRR using a routing policy

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. (Optional.) Disable LFA calculation on the interface.
   isis fast-reroute lfa-backup exclude [**level-1** | **level-2**]
   By default, the interface participates in LFA calculation, and can be elected as a backup interface.
4. Return to system view.
   quit
5. Enter IS-IS IPv4 unicast address family view.
   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]
   address-family ipv4 [**unicast**]
6. Enable IPv4 IS-IS FRR using a routing policy.
   - Create a routing policy and specify a backup next hop.
     apply fast-reroute backup-interface

For more information about the **apply fast-reroute backup-interface** command and routing policy, see *Layer 3—IP Routing Configuration Guide*.

- Configure IPv4 IS-IS FRR.

  fast-reroute route-policy *route-policy-name*

By default, IPv4 IS-IS FRR is disabled.

Enabling BFD control packet mode for IPv4 IS-IS FRR

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for IPv4 IS-IS FRR.

   isis primary-path-detect bfd ctrl

   By default, BFD control packet mode is disabled for IPv4 IS-IS FRR.

   To use BFD (control packet mode) to detect primary link failures, you must enable BFD control packet mode on both ends of the link.

Enabling BFD echo packet mode for IPv4 IS-IS FRR

1. Enter system view.

   system-view

2. Configure the source IP address of BFD echo packets.

   bfd echo-source-ip *ip-address*

   By default, the source IP address of BFD echo packets is not configured.

   The source IP address cannot be on the same network segment as any local interface's IP address.

   For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Enter interface view.

   interface *interface-type interface-number*

4. Enable BFD echo packet mode for IPv4 IS-IS FRR.

   isis primary-path-detect bfd echo

   By default, BFD echo packet mode is disabled for IPv4 IS-IS FRR.

   To use BFD (echo packet mode) to detect primary link failures, you only need to enable BFD echo packet mode on one end of the link.

Configuring IPv6 IS-IS FRR to calculate a backup next hop through LFA calculation

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. (Optional.) Disable LFA calculation on the interface.

   isis ipv6 fast-reroute lfa-backup exclude [**level-1** | **level-2**]

   By default, the interface participates in LFA calculation, and can be elected as a backup interface.

4. Return to system view.

   quit

5. Enter IS-IS view.

isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

6. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

7. Enable IPv6 IS-IS FRR to calculate a backup next hop through LFA calculation.

   fast-reroute lfa [**level-1** | **level-2**]

   By default, IPv6 IS-IS FRR is disabled.

Configuring IPv6 IS-IS FRR using a routing policy

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. (Optional.) Disable LFA calculation on the interface.

   isis ipv6 fast-reroute lfa-backup exclude [**level-1** | **level-2**]

   By default, the interface participates in LFA calculation, and can be elected as a backup interface.

4. Return to system view.

   quit

5. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

6. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

7. Enable IPv6 IS-IS FRR using a routing policy.

   - Create a routing policy and specify a backup next hop.

     apply ipv6 fast-reroute backup-interface

     For more information about the **apply ipv6 fast-reroute backup-interface** command and routing policy, see *Layer 3—IP Routing Configuration Guide*.

   - Configure IPv6 IS-IS FRR.

     fast-reroute route-policy *route-policy-name*

   By default, IPv6 IS-IS FRR is disabled.

Enabling BFD control packet mode for IPv6 IS-IS FRR

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD control packet mode for IPv6 IS-IS FRR.

   isis ipv6 primary-path-detect bfd ctrl

   By default, BFD control packet mode is disabled for IPv6 IS-IS FRR.

   To use BFD (control packet mode) to detect primary link failures, you must enable BFD control packet mode on both ends of the link.

Enabling BFD echo packet mode for IPv6 IS-IS FRR

1. Enter system view.

   system-view

2. Configure the source IPv6 address of BFD echo packets.

```
bfd echo-source-ipv6 ip-address
```

By default, the source IPv6 address of BFD echo packets is not configured.

The source IPv6 address cannot be on the same network segment as any local interface's IPv6 address.

For more information about this command, see *Network Management and Monitoring Command Reference*.

3. Enter interface view.

```
interface interface-type interface-number
```

4. Enable BFD echo packet mode for IPv6 IS-IS FRR.

```
isis ipv6 primary-path-detect bfd echo
```

By default, BFD echo packet mode is disabled for IPv6 IS-IS FRR.

To use BFD (echo packet mode) to detect primary link failures, you only need to enable BFD echo packet mode on one end of the link.

Setting the priority for FRR backup path selection policies

About this task

IS-IS FRR uses specific policies for backup path calculation. This command defines the priority for the backup path selection policy. The higher the value, the higher the priority of the associated backup path selection policy. Changing the backup path selection policy priority can affect the backup path calculation result for IS-IS FRR. The backup paths can provide node protection or link protection for traffic, or provide both node protection and link protection.

IS-IS FRR supports the following backup path selection policies that are used to generate different topologies for backup path calculation:

- **Node protection**—IS-IS FRR performs backup path calculation after excluding the primary next hop node.
- **Lowest cost**—IS-IS FRR performs backup path calculation after excluding the direct primary link.
- **SRLG disjoint**—When one link in the SRLG fails, the other links in the SRLG might also fail. If you use a link in this SRLG as the backup link for the failed link, protection does not take effect. To avoid this issue, IS-IS FRR excludes the local links in the same SRLG as the direct primary link and then performs backup path calculation.

For IS-IS FRR, the SRLG disjoint policy depends on the node protection and lowest cost policies.

If multiple backup path selection policies exist in an IS-IS process, the policy with the highest priority is used to calculate the backup path. If the policy fails to calculate the backup path, another policy with higher priority is used. IS-IS performs backup path calculation by using the node protection and lowest cost policies as follows:

- If the node protection policy has higher priority and fails to calculate the backup path, IS-IS uses the lowest cost policy to calculate the backup path. If the lowest cost policy still fails to calculate the backup path, reliability cannot be ensured upon primary link failure.
- If the lowest cost policy has higher priority and fails to calculate the backup path, IS-IS does not perform further backup path calculation with the node protection policy. Reliability cannot be ensured upon primary link failure.

Table 4 shows the backup path selection mechanism for IS-IS FRR based on priorities of backup path selection policies.

Table 4 Backup path selection mechanism for IS-IS FRR based on priorities of link selection policies

Priorities of link selection policies	Backup path selection mechanism for IS-IS FRR
Node protection > lowest cost > SRLG-disjoint	IS-IS FRR performs calculations based on the node protection and lowest cost policies in descending of priority. IS-IS FRR performs a maximum of two calculations. If IS-IS FRR calculates a backup path with a link selection policy, it does not perform further calculations.
Node protection > SRLG-disjoint > lowest cost	IS-IS FRR performs calculations based on the node protection, node protection + SRLG-disjoint, lowest cost + SRLG-disjoint, and lowest cost policies in descending of priority. IS-IS FRR performs a maximum of four calculations. If IS-IS FRR calculates a backup path with a link selection policy, it does not perform further calculations.
SRLG-disjoint > node protection > lowest cost	IS-IS FRR performs calculations based on the node protection + SRLG-disjoint, lowest cost + SRLG-disjoint, node protection, and lowest cost policies in descending of priority. IS-IS FRR performs a maximum of four calculations. If IS-IS FRR calculates a backup path with a link selection policy, it does not perform further calculations.
Lowest cost > node protection > SRLG-disjoint	IS-IS FRR performs calculations based on the lowest cost policy. IS-IS FRR performs only one calculation.
Lowest cost > SRLG-disjoint > node protection	IS-IS FRR performs calculations based on the lowest cost policy. IS-IS FRR performs only one calculation.
SRLG-disjoint > lowest cost > node protection	IS-IS FRR performs calculations based on the node protection + SRLG-disjoint, lowest cost + SRLG-disjoint, and lowest cost policies in descending of priority. IS-IS FRR performs a maximum of three calculations. If IS-IS FRR calculates a backup path with a link selection policy, it does not perform further calculations.

Setting the priority for IPv4 IS-IS FRR backup path selection policies

1. Enter system view.

   system-view

2. Enter IS-IS view.

   isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv4 address family view.

   address-family ipv4 [**unicast**]

4. Set the priority for the node-protection or lowest-cost backup path selection policy.

   fast-reroute tiebreaker { **lowest-cost** | **node-protecting** | **srlg-disjoint** } **preference** *preference* [**level-1** | **level-2**]

   By default, the priority values of the node-protection, lowest-cost, and shared risk link group (SRLG)-disjoint backup path selection policies are 40, 20, and 10, respectively.

Setting the priority for IPv6 IS-IS FRR backup path selection policies

1. Enter system view.

   system-view

2. Enter IS-IS view.

isis [*process-id*] [**vpn-instance** *vpn-instance-name*]

3. Enter IS-IS IPv6 address family view.

   address-family ipv6 [**unicast**]

4. Set the priority for the node-protection or lowest-cost backup path selection policy.

   fast-reroute tiebreaker { **lowest-cost** | **node-protecting** } **preference** *preference* [**level-1** | **level-2**]

   By default, the priority values of the node-protection and lowest-cost backup path selection policies are 40 and 20, respectively.

Display and maintenance commands for IS-IS

Execute **display** commands in any view and the **reset** command in user view.

Displaying and maintaining IPv4 IS-IS

Task	Command
Display IS-IS process information.	**display isis** [*process-id*]
Display IS-IS route calculation log information.	**display isis event-log spf** [**ipv4**] [[**level-1** | **level-2**] | **verbose**] * [*process-id*]
Display IS-IS GR log information.	**display isis graceful-restart event-log slot** *slot-number*
Display IS-IS GR status information.	**display isis graceful-restart status** [**level-1** | **level-2**] [*process-id*]
Display IS-IS interface information.	**display isis interface** [[*interface-type interface-number*] [**verbose**] | **statistics**] [*process-id*]
Display IS-IS LSDB information.	**display isis lsdb** [[**level-1** | **level-2**] | **local** | **lsp-id** *lspid* | [**lsp-name** *lspname*] | **verbose**] * [*process-id*]
Display IS-IS LSDB statistics.	**display isis lsdb statistics** [**level-1** | **level-2**] [*process-id*]
Display IS-IS system ID-to-host mapping information.	**display isis name-table** [*process-id*]
Display IS-IS NSR log information.	**display isis non-stop-routing event-log slot** *slot-number*
Display the IS-IS NSR status.	**display isis non-stop-routing status**
Display IS-IS neighbor information.	**display isis peer** [**statistics** | **verbose**] [*process-id*]
Display IPv4 IS-IS redistributed route information.	**display isis redistribute** [**ipv4** [*ip-address mask-length*]] [**level-1** | **level-2**] [*process-id*]
Display IS-IS IPv4 routing information.	**display isis route** [**ipv4** [*ip-address mask-length*]] [[**level-1** | **level-2**] | **verbose**] * [*process-id*]

Task	Command
Display IS-IS IPv4 topology information.	**display isis spf-tree** [**ipv4**] [[**level-1** | **level-2**] | [**source-id** *source-id* | **verbose**]] * [*process-id*]
Display IPv4 IS-IS statistics.	**display isis statistics** [**ipv4**] [**level-1** | **level-1-2** | **level-2**] [*process-id*]
Display OSI connection information.	**display osi** [**slot** *slot-number*]
Display OSI connection statistics.	**display osi statistics** [**slot** *slot-number*]
Clear IS-IS process data structure information.	**reset isis all** [*process-id*] [**graceful-restart**]
Clear IS-IS GR log information.	**reset isis graceful-restart event-log slot** *slot-number* **reset isis graceful-restart event-log chassis** *chassis-number* **slot** *slot-number*
Clear IS-IS NSR log information.	**reset isis non-stop-routing event-log slot** *slot-number*
Clear the data structure information of an IS-IS neighbor.	**reset isis peer** *system-id* [*process-id*]
Clear OSI connection statistics.	**reset osi statistics**

Displaying and maintaining IPv6 IS-IS

Task	Command
Display IS-IS process information.	**display isis** [*process-id*]
Display IS-IS route calculation log information.	**display isis event-log spf ipv6** [[**level-1** | **level-2**] | **verbose**] * [*process-id*]
Display IS-IS interface information.	**display isis interface** [[*interface-type interface-number*] [**verbose**] | **statistics**] [*process-id*]
Display IS-IS LSDB information.	**display isis lsdb** [[**level-1** | **level-2**] | **local** | **lsp-id** *lspid* | [**lsp-name** *lspname*] | **verbose**] * [*process-id*]
Display IS-IS LSDB statistics.	**display isis lsdb statistics** [**level-1** | **level-2**] [*process-id*]
Display the host name to system ID mapping table.	**display isis name-table** [*process-id*]
Display IS-IS neighbor information.	**display isis peer** [**statistics** | **verbose**] [*process-id*]
Display IPv6 IS-IS redistributed route information.	**display isis redistribute ipv6** [*ipv6-address mask-length*] [**level-1** | **level-2**] [*process-id*]
Display IPv6 IS-IS routing	**display isis route ipv6** [*ipv6-address*]

Task	Command
information.	[[**level-1** | **level-2**] | **verbose**] * [*process-id*]
Display IPv6 IS-IS topology information.	**display isis spf-tree ipv6** [[**level-1** | **level-2**] | [**source-id** *source-id* | **verbose**]] * [*process-id*]
Display IPv6 IS-IS statistics.	**display isis statistics ipv6** [**level-1** | **level-1-2** | **level-2**] [*process-id*]
Display OSI connection information.	**display osi** [**slot** *slot-number*]
Display OSI connection statistics.	**display osi statistics** [**slot** *slot-number*]
Clear IS-IS process data structure information.	**reset isis all** [*process-id*] [**graceful-restart**]
Clear the data structure information of an IS-IS neighbor.	**reset isis peer** *system-id* [*process-id*]
Clear OSI connection statistics.	**reset osi statistics**

IS-IS configuration examples

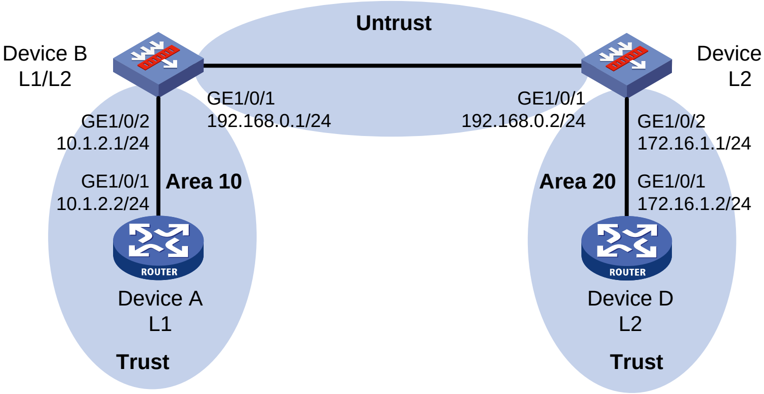
Example: Configuring basic IS-IS

Network configuration

As shown in Figure 9, Device A, Device B, Device C, and Device D reside in an AS.

Device A is a Level-1 device, Device C and Device D are Level-2 devices, and Device B is a Level-1-2 device connecting two areas. Device A and Device B are in area 10. Device C and Device D are in area 20.

Figure 9 Network diagram



Procedure

# Configure Device B.

1. Configure IP addresses for interfaces correctly according to the network diagram above.
   a. Configure an IP address for GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
```

```
[DeviceB-GigabitEthernet1/0/1] ip address 192.168.0.1 24
[DeviceB-GigabitEthernet1/0/1] quit
```

b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device B to a security zone.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.

a. Configure a security policy to allow IS-IS neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

# Create security policy rule **is-islocalin** and permit Device B to receive IS-IS packets from Device C.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name is-islocalin
[DeviceB-security-policy-ip-0-is-islocalin] source-zone untrust
[DeviceB-security-policy-ip-0-is-islocalin] destination-zone local
[DeviceB-security-policy-ip-0-is-islocalin] action pass
[DeviceB-security-policy-ip-0-is-islocalin] quit
```

# Create security policy rule **is-islocalout** and permit Device B to send IS-IS packets to Device C.

```
[DeviceB-security-policy-ip] rule name is-islocalout
[DeviceB-security-policy-ip-1-is-islocalout] source-zone local
[DeviceB-security-policy-ip-1-is-islocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-is-islocalout] action pass
[DeviceB-security-policy-ip-1-is-islocalout] quit
```

b. Configure a security policy to permit traffic between security zone **trust** and security zone **local**.

# Create security policy rule **trust-local** and permit Device A to send IS-IS packets to Device B.

```
[DeviceB-security-policy-ip] rule name trust-local
[DeviceB-security-policy-ip-2-trust-local] source-zone trust
[DeviceB-security-policy-ip-2-trust-local] destination-zone local
[DeviceB-security-policy-ip-2-trust-local] action pass
[DeviceB-security-policy-ip-2-trust-local] quit
```

# Create security policy rule **local-trust** and permit Device B to send IS-IS packets to Device A.

```
[DeviceB-security-policy-ip] rule name local-trust
[DeviceB-security-policy-ip-3-local-trust] source-zone local
[DeviceB-security-policy-ip-3-local-trust] destination-zone trust
[DeviceB-security-policy-ip-3-local-trust] action pass
[DeviceB-security-policy-ip-3-local-trust] quit
```

c. Configure a security policy that permits traffic between security zone **untrust** and security zone **trust** to permit traffic between area 10 and area 20.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-4-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-4-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-4-trust-untrust] source-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-4-trust-untrust] destination-ip-subnet 172.16.1.0 24
[DeviceB-security-policy-ip-4-trust-untrust] action pass
[DeviceB-security-policy-ip-4-trust-untrust] quit
```

# Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-5-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-5-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-5-untrust-trust] source-ip-subnet 172.16.1.0 24
[DeviceB-security-policy-ip-5-untrust-trust] destination-ip-subnet 10.1.2.0 24
[DeviceB-security-policy-ip-5-untrust-trust] action pass
[DeviceB-security-policy-ip-5-untrust-trust] quit
[DeviceB-security-policy-ip] quit
```

4. Enable IS-IS.

```
[DeviceB] isis 1
[DeviceB-isis-1] network-entity 10.0000.0000.0001.00
[DeviceB-isis-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] isis enable 1
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] isis enable 1
[DeviceB-GigabitEthernet1/0/2] quit
```

# Configure Device C.

1. Configure IP addresses for interfaces correctly according to the network diagram above.
   a. Configure an IP address for GigabitEthernet 1/0/1.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ip address 192.168.0.2 24
[DeviceC-GigabitEthernet1/0/1] quit
```

   b. Configure IP addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device C to a security zone.

```
[DeviceC] security-zone name untrust
[DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceC-security-zone-Untrust] quit
[DeviceC] security-zone name trust
[DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceC-security-zone-Trust] quit
```

3. Configure security policies.
   a. Configure a security policy to allow IS-IS neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

# Create security policy rule **is-islocalin** and permit Device C to receive IS-IS packets from Device B.

```
[DeviceC] security-policy ip
[DeviceC-security-policy-ip] rule name is-islocalin
[DeviceC-security-policy-ip-0-is-islocalin] source-zone untrust
[DeviceC-security-policy-ip-0-is-islocalin] destination-zone local
[DeviceC-security-policy-ip-0-is-islocalin] action pass
[DeviceC-security-policy-ip-0-is-islocalin] quit
```

# Create security policy rule **is-islocalout** and permit Device C to send IS-IS packets to Device B.

```
[DeviceC-security-policy-ip] rule name is-islocalout
[DeviceC-security-policy-ip-1-is-islocalout] source-zone local
[DeviceC-security-policy-ip-1-is-islocalout] destination-zone untrust
[DeviceC-security-policy-ip-1-is-islocalout] action pass
[DeviceC-security-policy-ip-1-is-islocalout] quit
```

b. Configure a security policy to permit traffic between security zone **trust** and security zone **local**.

# Create security policy rule **trust-local** and permit Device D to send IS-IS packets to Device C.

```
[DeviceC-security-policy-ip] rule name trust-local
[DeviceC-security-policy-ip-2-trust-local] source-zone trust
[DeviceC-security-policy-ip-2-trust-local] destination-zone local
[DeviceC-security-policy-ip-2-trust-local] action pass
[DeviceC-security-policy-ip-2-trust-local] quit
```

# Create security policy rule **local-trust** and permit Device C to send IS-IS packets to Device D.

```
[DeviceC-security-policy-ip] rule name local-trust
[DeviceC-security-policy-ip-3-local-trust] source-zone local
[DeviceC-security-policy-ip-3-local-trust] destination-zone trust
[DeviceC-security-policy-ip-3-local-trust] action pass
[DeviceC-security-policy-ip-3-local-trust] quit
```

c. Configure a security policy that permits traffic between security zone **untrust** and security zone **trust** to permit traffic between area 10 and area 20.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceC-security-policy-ip] rule name trust-untrust
[DeviceC-security-policy-ip-4-trust-untrust] source-zone trust
[DeviceC-security-policy-ip-4-trust-untrust] destination-zone untrust
[DeviceC-security-policy-ip-4-trust-untrust] source-ip-subnet 172.16.1.0 24
[DeviceC-security-policy-ip-4-trust-untrust] destination-ip-subnet 10.1.2.0 24
[DeviceC-security-policy-ip-4-trust-untrust] action pass
[DeviceC-security-policy-ip-4-trust-untrust] quit
```

# Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceC-security-policy-ip] rule name untrust-trust
[DeviceC-security-policy-ip-5-untrust-trust] source-zone untrust
[DeviceC-security-policy-ip-5-untrust-trust] destination-zone trust
[DeviceC-security-policy-ip-5-untrust-trust] source-ip-subnet 10.1.2.0 24
```

```
[DeviceC-security-policy-ip-5-untrust-trust] destination-ip-subnet 172.16.1.0
24
[DeviceC-security-policy-ip-5-untrust-trust] action pass
[DeviceC-security-policy-ip-5-untrust-trust] quit
[DeviceC-security-policy-ip] quit
```

4. Enable IS-IS.

```
[DeviceC] isis 1
[DeviceC-isis-1] is-level level-2
[DeviceC-isis-1] network-entity 20.0000.0000.0001.00
[DeviceC-isis-1] quit
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] isis enable 1
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] isis enable 1
[DeviceC-GigabitEthernet1/0/2] quit
```

# Configure Device A.

1. Configure an IP address for GigabitEthernet 1/0/1 correctly according to the network diagram above.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.2 24
[DeviceA-GigabitEthernet1/0/1] quit
```

2. Enable IS-IS.

```
[DeviceA] isis 1
[DeviceA-isis-1] is-level level-1
[DeviceA-isis-1] network-entity 10.0000.0000.0002.00
[DeviceA-isis-1] quit
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] isis enable 1
[DeviceA-GigabitEthernet1/0/1] quit
```

# Configure Device D.

1. Configure an IP address for GigabitEthernet 1/0/1 correctly according to the network diagram above.

```
<DeviceD> system-view
[DeviceD] interface gigabitethernet 1/0/1
[DeviceD-GigabitEthernet1/0/1] ip address 172.16.1.2 24
[DeviceD-GigabitEthernet1/0/1] quit
```

2. Enable IS-IS.

```
[DeviceD] isis 1
[DeviceD-isis-1] is-level level-2
[DeviceD-isis-1] network-entity 20.0000.0000.0002.00
[DeviceD-isis-1] quit
[DeviceD] interface gigabitethernet 1/0/1
[DeviceD-GigabitEthernet1/0/1] isis enable 1
[DeviceD-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# View IS-IS routing information on Device B.

```
[Device B] display isis route

                         Route information for IS-IS(1)
                         -----------------------------

                         Level-1 IPv4 Forwarding Table
                         -----------------------------

 IPv4 Destination     IntCost    ExtCost ExitInterface   NextHop         Flags
-------------------------------------------------------------------------------
 10.1.2.0/24          10         NULL    GE1/0/2         Direct          D/L/-
 192.168.0.0/24       10         NULL    GE1/0/1         Direct          D/L/-

      Flags: D-Direct, R-Added to Rib, L-Advertised in LSPs, U-Up/Down Bit Set

                         Level-2 IPv4 Forwarding Table
                         -----------------------------

 IPv4 Destination     IntCost    ExtCost ExitInterface   NextHop         Flags
-------------------------------------------------------------------------------
 10.1.2.0/24          10         NULL                                    D/L/-
 192.168.0.0/24       10         NULL                                    D/L/-
 172.16.0.0/16        20         NULL    GE1/0/2         192.168.0.2     R/-/-

      Flags: D-Direct, R-Added to Rib, L-Advertised in LSPs, U-Up/Down Bit Set
```

# Verify that devices in area 10 can ping the devices in area 20, and vice versa.

IPv6 IS-IS configuration examples

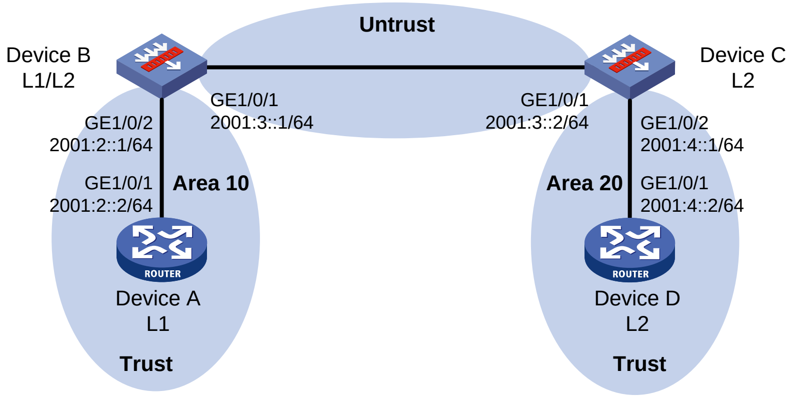
Example: Configuring IPv6 IS-IS basics

Network configuration

As shown in Figure 10, Device A, Device B, Device C, and Device D reside in an AS.

Device A is a Level-1 device, Device C and Device D are Level-2 devices, and Device B is a Level-1-2 device connecting two areas. Device A and Device B are in area 10. Device C and Device D are in area 20.

Figure 10 Network diagram



Procedure

# Configure Device B.

1. Configure IPv6 addresses for interfaces correctly according to the network diagram above.
   a. Configure an IPv6 address for GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipv6 address 2001:3::1 64
[DeviceB-GigabitEthernet1/0/1] quit
```

   b. Configure IPv6 addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)

2. Add each interface on Device B to a security zone.

```
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.
   a. Configure a security policy to allow IPv6 IS-IS neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **is-islocalin** and permit Device B to receive IPv6 IS-IS packets from Device C.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name is-islocalin
[DeviceB-security-policy-ipv6-0-is-islocalin] source-zone untrust
[DeviceB-security-policy-ipv6-0-is-islocalin] destination-zone local
[DeviceB-security-policy-ipv6-0-is-islocalin] action pass
[DeviceB-security-policy-ipv6-0-is-islocalin] quit
```

   # Create security policy rule **is-islocalout** and permit Device B to send IPv6 IS-IS packets to Device C.

```
[DeviceB-security-policy-ipv6] rule name is-islocalout
[DeviceB-security-policy-ipv6-1-is-islocalout] source-zone local
[DeviceB-security-policy-ipv6-1-is-islocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-is-islocalout] action pass
[DeviceB-security-policy-ipv6-1-is-islocalout] quit
```

b. Configure a security policy to permit traffic between security zone **trust** and security zone **local**.

   # Create security policy rule **trust-local** and permit Device A to send IPv6 IS-IS packets to Device B.

   ```
   [DeviceB-security-policy-ipv6] rule name trust-local
   [DeviceB-security-policy-ipv6-2-trust-local] source-zone trust
   [DeviceB-security-policy-ipv6-2-trust-local] destination-zone local
   [DeviceB-security-policy-ipv6-2-trust-local] action pass
   [DeviceB-security-policy-ipv6-2-trust-local] quit
   ```

   # Create security policy rule **local-trust** and permit Device B to send IPv6 IS-IS packets to Device A.

   ```
   [DeviceB-security-policy-ipv6] rule name local-trust
   [DeviceB-security-policy-ipv6-3-local-trust] source-zone local
   [DeviceB-security-policy-ipv6-3-local-trust] destination-zone trust
   [DeviceB-security-policy-ipv6-3-local-trust] action pass
   [DeviceB-security-policy-ipv6-3-local-trust] quit
   [DeviceB-security-policy-ipv6] quit
   ```

c. Configure a security policy that permits traffic between security zone **untrust** and security zone **trust** to permit traffic between area 10 and area 20.

   # Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

   ```
   [DeviceB-security-policy-ipv6] rule name trust-untrust
   [DeviceB-security-policy-ipv6-4-trust-untrust] source-zone trust
   [DeviceB-security-policy-ipv6-4-trust-untrust] destination-zone untrust
   [DeviceB-security-policy-ipv6-4-trust-untrust] source-ip-subnet 2001:2:: 64
   [DeviceB-security-policy-ipv6-4-trust-untrust] destination-ip-subnet 2001:4::
   64
   [DeviceB-security-policy-ipv6-4-trust-untrust] action pass
   [DeviceB-security-policy-ipv6-4-trust-untrust] quit
   ```

   # Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

   ```
   [DeviceB-security-policy-ipv6] rule name untrust-trust
   [DeviceB-security-policy-ipv6-5-untrust-trust] source-zone untrust
   [DeviceB-security-policy-ipv6-5-untrust-trust] destination-zone trust
   [DeviceB-security-policy-ipv6-5-untrust-trust] source-ip-subnet 2001:4:: 64
   [DeviceB-security-policy-ipv6-5-untrust-trust] destination-ip-subnet 2001:2::
   64
   [DeviceB-security-policy-ipv6-5-untrust-trust] action pass
   [DeviceB-security-policy-ipv6-5-untrust-trust] quit
   [DeviceB-security-policy-ipv6] quit
   ```

4. Enable IS-IS.

```
[DeviceB] isis 1
[DeviceB-isis-1] network-entity 10.0000.0000.0001.00
[DeviceB-isis-1] address-family ipv6
[DeviceB-isis-1-ipv6] quit
[DeviceB-isis-1] quit
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] isis ipv6 enable 1
[DeviceB-GigabitEthernet1/0/1] quit
```

```
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] isis ipv6 enable 1
[DeviceB-GigabitEthernet1/0/2] quit
```

# Configure Device C.

1. Configure IPv6 addresses for interfaces correctly according to the network diagram above.
   a. Configure an IPv6 address for GigabitEthernet 1/0/1.
   ```
   <DeviceC> system-view
   [DeviceC] interface gigabitethernet 1/0/1
   [DeviceC-GigabitEthernet1/0/1] ipv6 address 2001:3::2 64
   [DeviceC-GigabitEthernet1/0/1] quit
   ```
   b. Configure IPv6 addresses for other interfaces in the same way GigabitEthernet 1/0/1 is configured. (Details not shown.)
2. Add each interface on Device C to a security zone.
   ```
   [DeviceC] security-zone name untrust
   [DeviceC-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceC-security-zone-Untrust] quit
   [DeviceC] security-zone name trust
   [DeviceC-security-zone-Trust] import interface gigabitethernet 1/0/2
   [DeviceC-security-zone-Trust] quit
   ```
3. Configure security policies.
   a. Configure a security policy to allow IS-IS neighbor relationship establishment by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **is-islocalin** and permit Device C to receive IPv6 IS-IS packets from Device B.
   ```
   [DeviceC] security-policy ipv6
   [DeviceC-security-policy-ipv6] rule name is-islocalin
   [DeviceC-security-policy-ipv6-0-is-islocalin] source-zone untrust
   [DeviceC-security-policy-ipv6-0-is-islocalin] destination-zone local
   [DeviceC-security-policy-ipv6-0-is-islocalin] action pass
   [DeviceC-security-policy-ipv6-0-is-islocalin] quit
   ```
   # Create security policy rule **is-islocalout** and permit Device C to send IPv6 IS-IS packets to Device B.
   ```
   [DeviceC-security-policy-ipv6] rule name is-islocalout
   [DeviceC-security-policy-ipv6-1-is-islocalout] source-zone local
   [DeviceC-security-policy-ipv6-1-is-islocalout] destination-zone untrust
   [DeviceC-security-policy-ipv6-1-is-islocalout] action pass
   [DeviceC-security-policy-ipv6-1-is-islocalout] quit
   ```
   b. Configure a security policy to permit traffic between security zone **trust** and security zone **local**.

   # Create security policy rule **trust-local** and permit Device D to send IPv6 IS-IS packets to Device C.
   ```
   [DeviceC-security-policy-ipv6] rule name trust-local
   [DeviceC-security-policy-ipv6-2-trust-local] source-zone trust
   [DeviceC-security-policy-ipv6-2-trust-local] destination-zone local
   [DeviceC-security-policy-ipv6-2-trust-local] action pass
   [DeviceC-security-policy-ipv6-2-trust-local] quit
   ```
   # Create security policy rule **local-trust** and permit Device C to send IPv6 IS-IS packets to Device D.

```
[DeviceC-security-policy-ipv6] rule name local-trust
[DeviceC-security-policy-ipv6-3-local-trust] source-zone local
[DeviceC-security-policy-ipv6-3-local-trust] destination-zone trust
[DeviceC-security-policy-ipv6-3-local-trust] action pass
[DeviceC-security-policy-ipv6-3-local-trust] quit
```

c. Configure a security policy that permits traffic between security zone **untrust** and security zone **trust** to permit traffic between area 10 and area 20.

   # Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceC-security-policy-ipv6] rule name trust-untrust
[DeviceC-security-policy-ipv6-4-trust-untrust] source-zone trust
[DeviceC-security-policy-ipv6-4-trust-untrust] destination-zone untrust
[DeviceC-security-policy-ipv6-4-trust-untrust] source-ip-subnet 2001:4:: 64
[DeviceC-security-policy-ipv6-4-trust-untrust] destination-ip-subnet 2001:2::
64
[DeviceC-security-policy-ipv6-4-trust-untrust] action pass
[DeviceC-security-policy-ipv6-4-trust-untrust] quit
```

   # Create security policy rule **untrust-trust** and permit packets from security zone **untrust** to security zone **trust** to pass.

```
[DeviceC-security-policy-ipv6] rule name untrust-trust
[DeviceC-security-policy-ipv6-5-untrust-trust] source-zone untrust
[DeviceC-security-policy-ipv6-5-untrust-trust] destination-zone trust
[DeviceC-security-policy-ipv6-5-untrust-trust] source-ip-subnet 2001:2:: 64
[DeviceC-security-policy-ipv6-5-untrust-trust] destination-ip-subnet 2001:4::
64
[DeviceC-security-policy-ipv6-5-untrust-trust] action pass
[DeviceC-security-policy-ipv6-5-untrust-trust] quit
[DeviceC-security-policy-ipv6] quit
```

4. Enable IPv6 IS-IS.

```
[DeviceC] isis 1
[DeviceC-isis-1] is-level level-2
[DeviceC-isis-1] network-entity 20.0000.0000.0001.00
[DeviceC-isis-1] address-family ipv6
[DeviceC-isis-1-ipv6] quit
[DeviceC-isis-1] quit
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] isis ipv6 enable 1
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] isis ipv6 enable 1
[DeviceC-GigabitEthernet1/0/2] quit
```

# Configure Device A.

1. Configure an IPv6 address for GigabitEthernet 1/0/1 correctly according to the network diagram above.

```
<DeviceA> system-view
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipv6 address 2001:2::2 64
[DeviceA-GigabitEthernet1/0/1] quit
```

2. Enable IPv6 IS-IS.

```
[DeviceA] isis 1
[DeviceA-isis-1] is-level level-1
[DeviceA-isis-1] network-entity 10.0000.0000.0002.00
[DeviceA-isis-1] address-family ipv6
[DeviceA-isis-1-ipv6] quit
[DeviceA-isis-1] quit
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] isis ipv6 enable 1
[DeviceA-GigabitEthernet1/0/1] quit
```

# Configure Device D.

1. Configure an IPv6 address for GigabitEthernet 1/0/1 correctly according to the network diagram above.

```
<DeviceD> system-view
[DeviceD] interface gigabitethernet 1/0/1
[DeviceD-GigabitEthernet1/0/1] ipv6 address 2001:4::2 64
[DeviceD-GigabitEthernet1/0/1] quit
```

2. Enable IPv6 IS-IS.

```
[DeviceD] isis 1
[DeviceD-isis-1] is-level level-2
[DeviceD-isis-1] network-entity 20.0000.0000.0002.00
[DeviceD-isis-1] address-family ipv6
[DeviceD-isis-1-ipv6] quit
[DeviceD-isis-1] quit
[DeviceD] interface gigabitethernet 1/0/1
[DeviceD-GigabitEthernet1/0/1] isis ipv6 enable 1
[DeviceD-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# View IPv6 IS-IS routing information on Device B.

```
[DeviceB] display isis route ipv6

                         Route information for IS-IS(1)
                         ------------------------------

                         Level-1 IPv6 forwarding table
                         -----------------------------

 Destination : 2001:2::                          PrefixLen: 64
 Flag        : D/L/-                             Cost     : 10
 Next hop    : Direct                            Interface: GE1/0/2

 Destination : 2001:3::                          PrefixLen: 64
 Flag        : D/L/-                             Cost     : 10
 Next hop    : Direct                            Interface: GE1/0/1
      Flags: D-Direct, R-Added to Rib, L-Advertised in LSPs, U-Up/Down Bit Set

                         Level-2 IPv6 forwarding table
                         -----------------------------
```

```
Destination : 2001:2::                                          PrefixLen: 64
Flag        : D/L/-                                             Cost     : 10

Destination : 2001:3::                                          PrefixLen: 64
Flag        : D/L/-                                             Cost     : 10

Destination : 2001:4::1                                         PrefixLen: 64
Flag        : R/-/-                                             Cost     : 10

     Flags: D-Direct, R-Added to Rib, L-Advertised in LSPs, U-Up/Down Bit Set
```

# Verify that devices in area 10 can ping the devices in area 20, and vice versa.

Contents

BGP overview

Border Gateway Protocol (BGP) is an exterior gateway protocol (EGP). It is called internal BGP (IBGP) when it runs within an AS and called external BGP (EBGP) when it runs between ASs. The current version in use is BGP-4 (RFC 4271).

BGP characteristics

BGP has the following characteristics:

- Focuses on route control and selection rather than route discovery and calculation.
- Uses TCP to enhance reliability.
- Measures the distance of a route by using a list of ASs that the route must travel through to reach the destination. BGP is also called a path-vector protocol.
- Supports CIDR.
- Reduces bandwidth consumption by advertising only incremental updates. BGP is very suitable to advertise large numbers of routes on the Internet.
- Eliminates routing loops by adding AS path information to BGP route updates.
- Uses policies to implement flexible route filtering and selection.
- Has good scalability.

BGP speaker and BGP peer

A router running BGP is a BGP speaker. A BGP speaker establishes peer relationships with other BGP speakers to exchange routing information over TCP connections.

BGP peers include the following types:

- **IBGP peers**—Reside in the same AS as the local router.
- **EBGP peers**—Reside in different ASs from the local router.

BGP message types

BGP uses the following message types:

- **Open**—After establishing a TCP connection, BGP sends an OPEN message to establish a session to the peer.
- **Update**—BGP sends UPDATE messages to exchange routing information between peers. Each UPDATE message can advertise a group of feasible routes with identical attributes and multiple withdrawn routes.
- **Keepalive**—BGP sends KEEPALIVE messages between peers to maintain connectivity.
- **Route-refresh**—BGP sends a ROUTE-REFRESH message to request the routing information for a specific address family from a peer.
- **Notification**—BGP sends a NOTIFICATION message upon detecting an error and immediately closes the connection.

BGP path attributes

BGP uses the following path attributes in UPDATE messages for route filtering and selection:

ORIGIN

The ORIGIN attribute specifies the origin of BGP routes. This attribute has the following types:

- **IGP**—Has the highest priority. Routes generated in the local AS have the IGP attribute.
- **EGP**—Has the second highest priority. Routes obtained through EGP have the EGP attribute.
- **INCOMPLETE**—Has the lowest priority. The source of routes with this attribute is unknown. Routes redistributed from other routing protocols have the INCOMPLETE attribute.

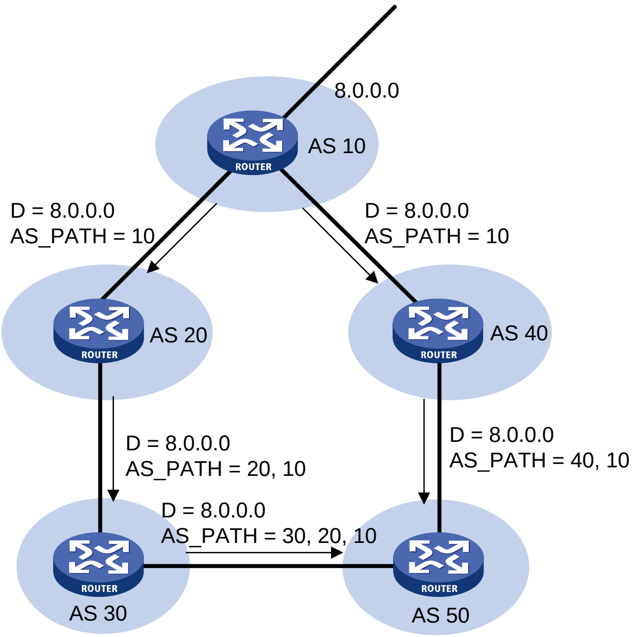
AS_PATH

The AS_PATH attribute identifies the ASs through which a route has passed. Before advertising a route to another AS, BGP adds the local AS number into the AS_PATH attribute, so the receiver can determine ASs to route the message back.

The AS_PATH attribute has the following types:

- **AS_SEQUENCE**—Arranges AS numbers in sequence. As shown in Figure 1, the number of the AS closest to the receiver's AS is leftmost.
- **AS_SET**—Arranges AS numbers randomly.

Figure 1 AS_PATH attribute



BGP uses the AS_PATH attribute to implement the following functions:

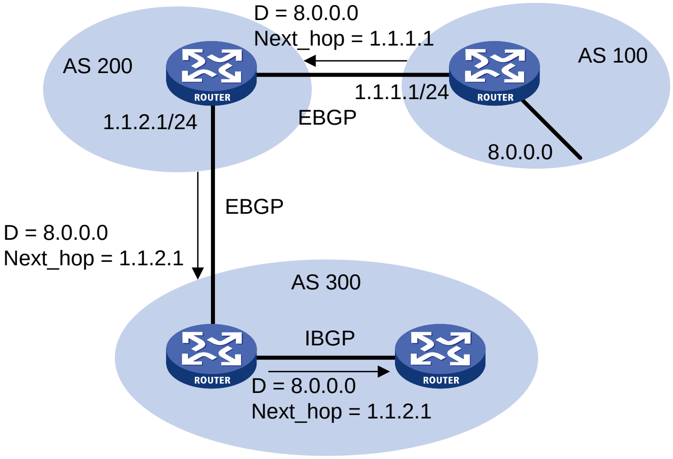
- **Avoid routing loops**—A BGP router does not receive routes containing the local AS number to avoid routing loops.
- **Affect route selection**—BGP gives priority to the route with the shortest AS_PATH length if other factors are the same. As shown in Figure 1, the BGP router in AS 50 gives priority to the route passing AS 40 for sending data to the destination 8.0.0.0. In some applications, you can apply a routing policy to control BGP route selection by modifying the AS_PATH length. For more information about routing policy, see "Configuring routing policies."
- **Filter routes**—By using an AS path list, you can filter routes based on AS numbers contained in the AS_PATH attribute. For more information about AS path list, see "Configuring routing policies."

NEXT_HOP

The NEXT_HOP attribute may not be the IP address of a directly connected router. Its value is determined as follows:

- When a BGP speaker advertises a self-originated route to a BGP peer, it sets the address of the sending interface as the NEXT_HOP.
- When a BGP speaker sends a received route to an EBGP peer, it sets the address of the sending interface as the NEXT_HOP.
- When a BGP speaker sends a route received from an EBGP peer to an IBGP peer, it does not modify the NEXT_HOP attribute. If load balancing is configured, BGP modifies the NEXT_HOP attribute for the equal-cost routes. For load balancing information, see "BGP load balancing."

Figure 2 NEXT_HOP attribute

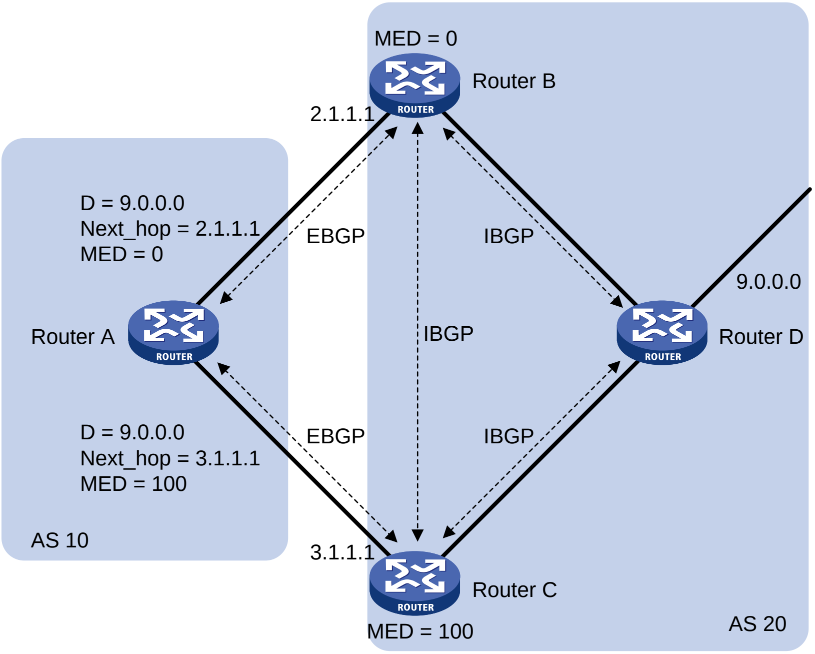


MED (MULTI_EXIT_DISC)

BGP advertises the MED attribute between two neighboring ASs, each of which does not advertise the attribute to any other AS.

Similar to metrics used by IGPs, MED is used to determine the optimal route for traffic going into an AS. When a BGP router obtains multiple routes to the same destination but with different next hops, it selects the route with the smallest MED value as the optimal route. As shown in Figure 3, traffic from AS 10 to AS 20 travels through Router B that is selected according to MED.

Figure 3 MED attribute



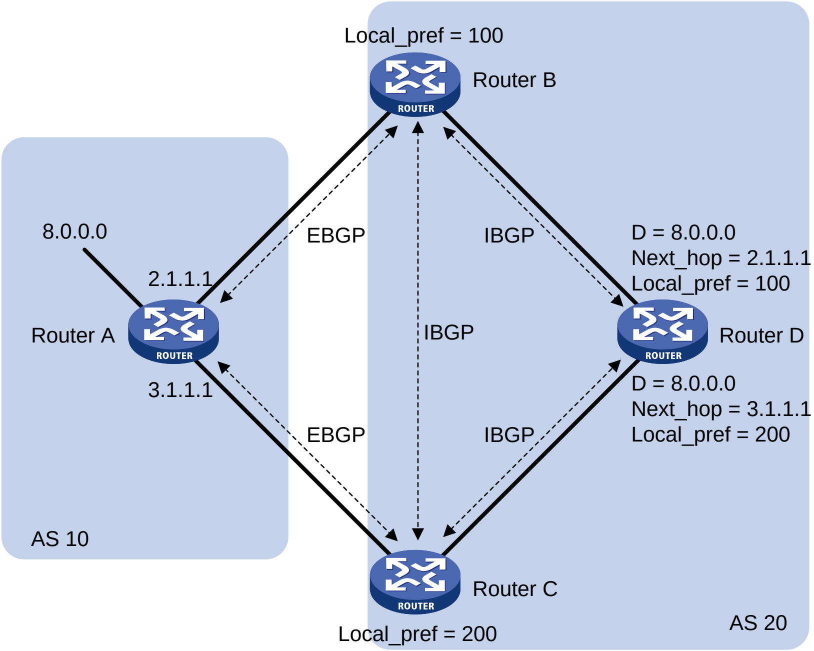
Generally BGP only compares MEDs of routes received from the same AS. You can also use the **compare-different-as-med** command to force BGP to compare MED values of routes received from different ASs.

LOCAL_PREF

The LOCAL_PREF attribute is exchanged between IBGP peers only, and is not advertised to any other AS. It indicates the priority of a BGP router.

BGP uses LOCAL_PREF to determine the optimal route for traffic leaving the local AS. When a BGP router obtains multiple routes to the same destination but with different next hops, it selects the route with the highest LOCAL_PREF value as the optimal route. As shown in Figure 4, traffic from AS 20 to AS 10 travels through Router C that is selected according to LOCAL_PREF.

Figure 4 LOCAL_PREF attribute



COMMUNITY

The COMMUNITY attribute identifies the community of BGP routes. A BGP community is a group of routes with the same characteristics. It has no geographical boundaries. Routes of different ASs can belong to the same community.

A route can carry one or more COMMUNITY attribute values (each of which is represented by a 4-byte integer). A router uses the COMMUNITY attribute to determine whether to advertise the route and the advertising scope without using complex filters such as ACLs. This mechanism simplifies routing policy configuration, management, and maintenance.

Well-known COMMUNITY attributes involve the following:

- **INTERNET**—By default, all routes belong to the Internet community. Routes with this attribute can be advertised to all BGP peers.
- **NO_EXPORT**—Routes with this attribute cannot be advertised out of the local AS or out of the local confederation, but can be advertised to other sub-ASs in the confederation. For confederation information, see "Settlements for problems in large-scale BGP networks."
- **No_ADVERTISE**—Routes with this attribute cannot be advertised to other BGP peers.
- **No_EXPORT_SUBCONFED**—Routes with this attribute cannot be advertised out of the local AS or other sub-ASs in the local confederation.

You can configure BGP community lists to filter BGP routes based on the BGP COMMUNITY attribute.

Extended community attribute

To meet new demands, BGP defines the extended community attribute. The extended community attribute has the following advantages over the COMMUNITY attribute:

- Provides more attribute values by extending the attribute length to eight bytes.
- Allows for using different types of extended community attributes in different scenarios to enhance route filtering and control and simplify configuration and management.

The device supports the route target attribute and Site of Origin (SoO) extended community attribute.

The SoO attribute specifies the site where the route originated. It prevents advertising a route back to the originating site. If the AS-path attribute is lost, the router can use the SoO attribute to avoid routing loops.

The SoO attribute has the following formats:

- *16-bit AS number:32-bit user-defined number*. For example, 100:3.
- *32-bit IP address:16-bit user-defined number*. For example, 192.168.122.15:1.
- *32-bit AS number:16-bit user-defined number*, where the minimum value of the AS number is 65536. For example, 65536:1.

BGP route selection

BGP discards routes with unreachable NEXT_HOPs. If multiple routes to the same destination are available, BGP selects the optimal route in the following sequence:

1. The route with the highest Preferred_value.
2. The route with the highest LOCAL_PREF.
3. The route generated by the **network** command, the route redistributed by the **import-route** command, or the summary route in turn.
4. The route with the shortest AS_PATH.
5. The IGP, EGP, or INCOMPLETE route in turn.
6. The route with the lowest MED value.
7. The route learned from EBGP, confederation EBGP, confederation IBGP, or IBGP in turn.
8. The route with the smallest IGP metric.
9. The route with the smallest recursion depth.
10. If a route received from an EBGP peer is the current optimal route, BGP does not change the optimal route when it receives routes from other EBGP peers.
11. The route advertised by the router with the smallest router ID.

    If one of the routes is advertised by a route reflector, BGP compares the ORIGINATOR_ID of the route with the router IDs of other routers. Then, BGP selects the route with the smallest ID as the optimal route.
12. The route with the shortest CLUSTER_LIST.
13. The route advertised by the peer with the lowest IP address.

The CLUSTER_IDs of route reflectors form a CLUSTER_LIST. If a route reflector receives a route that contains its own CLUSTER ID in the CLUSTER_LIST, the router discards the route to avoid routing loops.

If load balancing is configured, the system selects available routes to implement load balancing.

BGP route advertisement rules

BGP follows these rules for route advertisement:

- When multiple feasible routes to a destination exist, BGP advertises only the optimal route to its peers. If the **`advertise-rib-active`** command is configured, BGP advertises the optimal route in the IP routing table. If not, BGP advertises the optimal route in the BGP routing table.
- BGP advertises only routes that it uses.
- BGP advertises routes learned from an EBGP peer to all BGP peers, including both EBGP and IBGP peers.
- BGP advertises routes learned from an IBGP peer to EBGP peers, rather than other IBGP peers.
- After establishing a session to a new BGP peer, BGP advertises all the routes matching the above rules to the peer. After that, BGP advertises only incremental updates to the peer.

BGP load balancing

BGP load balancing is applicable between EBGP peers, between IBGP peers, and between confederations.

BGP implements load balancing through route recursion and route selection.

BGP load balancing through route recursion

The next hop of a BGP route might not be directly connected. One of the reasons is that the next hop information exchanged between IBGP peers is not modified. The BGP router must find the directly connected next hop through IGP. The matching route with the direct next hop is called the recursive route. The process of finding a recursive route is route recursion.

If multiple recursive routes to the same destination are load balanced, BGP generates the same number of next hops to forward packets.

BGP load balancing based on route recursion is always enabled in the system.

BGP load balancing through route selection

IGP routing protocols, such as RIP and OSPF, can use route metrics as criteria to load balance between routes that have the same metric. BGP cannot load balance between routes by route metrics as an IGP protocol does, because BGP does not have a route computation algorithm.

BGP uses the following load balancing criteria to determine load balanced routes:

- The routes have the same ORIGIN, LOCAL_PREF, and MED attributes.
- The routes meet the following requirements on the AS_PATH attribute:
    - If the **`balance as-path-neglect`** command is configured, the routes can have different AS_PATH attributes.
    - If only the **`balance as-path-relax`** command is configured, the routes can have different AS_PATH attributes, but the length of the AS_PATH attributes must be the same.
    - If neither the **`balance as-path-neglect`** nor the **`balance as-path-relax`** command is configured, the routes must have the same AS_PATH attribute.
- The next hops of the routes meet the following requirements on IGP metrics:
    - If the **`bestroute igp-metric-ignore`** command is not configured, the next hops of the routes must have the same IGP metric value.
    - If the **`bestroute igp-metric-ignore`** command is configured, the next hops of the routes can have different IGP metric values.

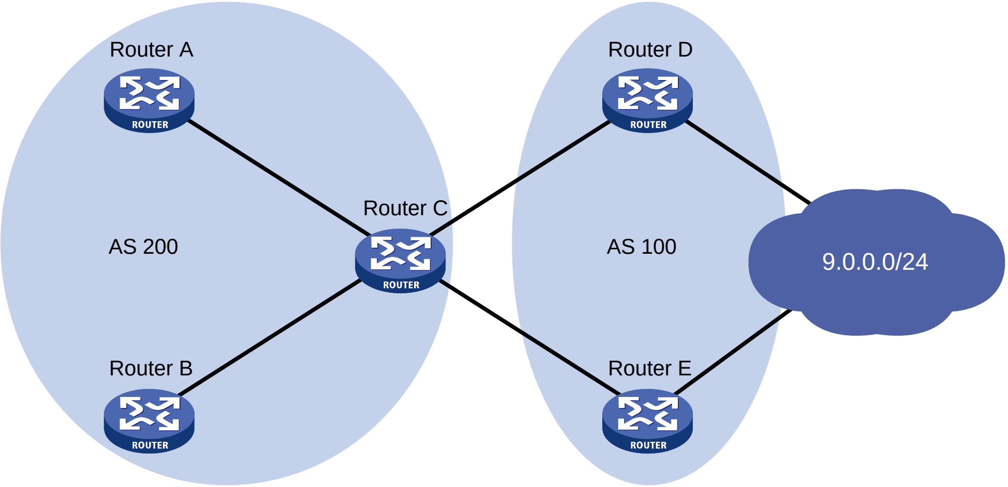
BGP does not use the route selection rules described in "BGP route selection" for load balancing.

As shown in Figure 5, Router A and Router B are IBGP peers of Router C. Router C allows a maximum number of two ECMP routes for load balancing.

Router D and Router E both advertise a route 9.0.0.0 to Router C. Router C installs the two routes to its routing table for load balancing if the routes meet the BGP load balancing criteria. After that, Router C forwards to Router A and Router B a single route whose attributes are changed as follows:

- AS_PATH attribute:
  - If the **`balance as-path-neglect`** and **`balance as-path-relax`** commands are not configured, the AS_PATH attribute does not change.
  - If the **`balance as-path-neglect`** or **`balance as-path-relax`** command is configured, the AS_PATH attribute is changed to the attribute of the optimal route.
- The NEXT_HOP attribute is changed to the IP address of Router C.
- Other attributes are changed to be the same as the optimal route.

Figure 5 Network diagram



Settlements for problems in large-scale BGP networks

You can use the following methods to facilitate management and improve route distribution efficiency on a large-scale BGP network.

Route summarization

Route summarization can reduce the BGP routing table size by advertising summary routes rather than more specific routes.

The system supports both manual and automatic route summarization. Manual route summarization allows you to determine the attribute of a summary route and whether to advertise more specific routes.

Route dampening

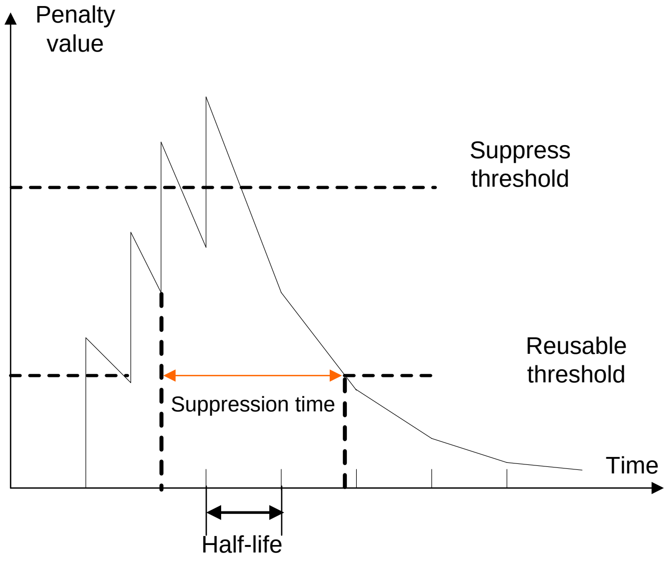
Route flapping (a route comes up and disappears in the routing table frequently) causes BGP to send many routing updates. It can consume too many resources and affect other operations.

In most cases, BGP runs in complex networks where route changes are more frequent. To solve the problem caused by route flapping, you can use BGP route dampening to suppress unstable routes.

BGP route dampening uses a penalty value to judge the stability of a route. The bigger the value, the less stable the route. Each time a route state changes from reachable to unreachable, or a reachable route's attribute changes, BGP adds a penalty value of 1000 to the route. When the penalty value of the route exceeds the suppress value, the route is suppressed and cannot become the optimal route. When the penalty value reaches the upper limit, no penalty value is added.

If the suppressed route does not flap, its penalty value gradually decreases to half of the suppress value after a period of time. This period is called "Half-life." When the value decreases to the reusable threshold value, the route is usable again.

Figure 6 BGP route dampening



Peer group

You can organize BGP peers with the same attributes into a group to simplify their configurations.

When a peer joins the peer group, the peer obtains the same configuration as the peer group. If the configuration of the peer group is changed, the configuration of group members is changed.

Community

You can apply a community list or an extended community list to a routing policy for route control. For more information, see "BGP path attributes."

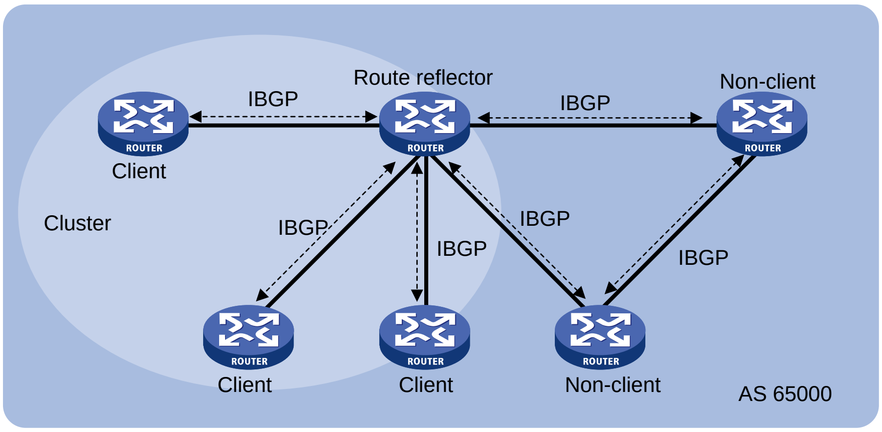
Route reflector

IBGP peers must be fully meshed to maintain connectivity. If n routers exist in an AS, the number of IBGP connections is n(n-1)/2. If a large number of IBGP peers exist, large amounts of network and CPU resources are consumed to maintain sessions.

Using route reflectors can solve this issue. In an AS, a router acts as a route reflector, and other routers act as clients connecting to the route reflector. The route reflector forwards routing information received from a client to other clients. In this way, all clients can receive routing information from one another without establishing BGP sessions.

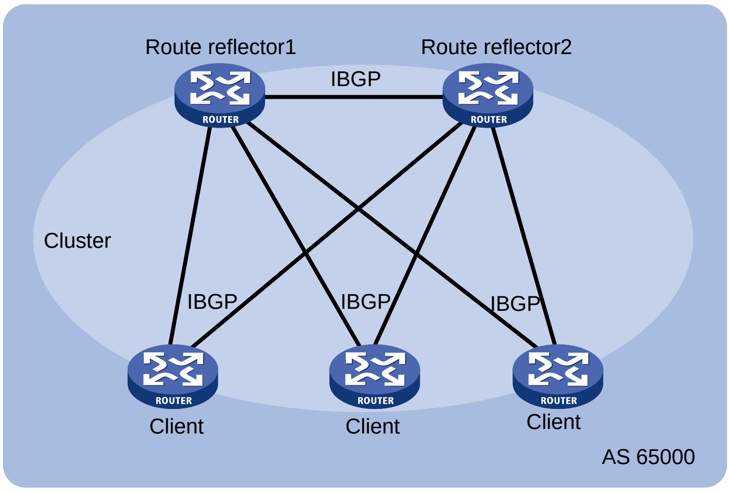
A router that is neither a route reflector nor a client is a non-client, which, as shown in Figure 7, must establish BGP sessions to the route reflector and other non-clients.

Figure 7 Network diagram for a route reflector



The route reflector and clients form a cluster. Typically a cluster has one route reflector. The ID of the route reflector is the Cluster_ID. You can configure more than one route reflector in a cluster to improve availability, as shown in Figure 8. The configured route reflectors must have the same Cluster_ID to avoid routing loops.

Figure 8 Network diagram for route reflectors



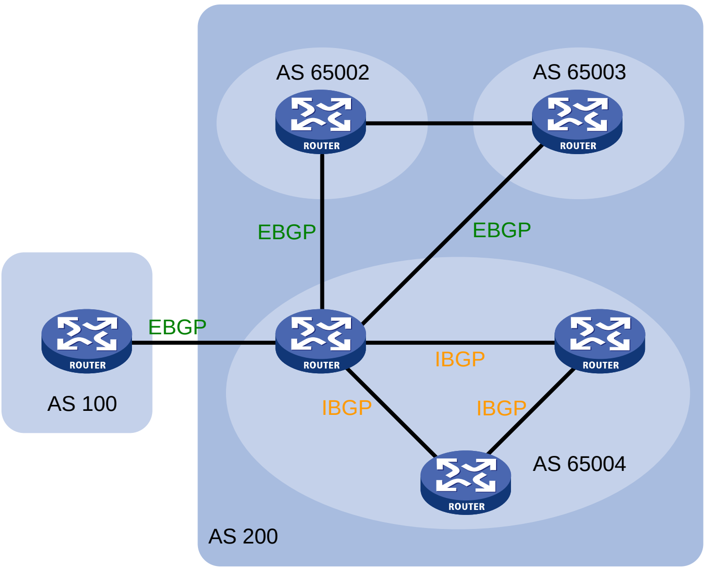
When the BGP routers in an AS are fully meshed, route reflection is unnecessary because it consumes more bandwidth resources. You can use commands to disable route reflection instead of modifying network configuration or changing network topology.

After route reflection is disabled between clients, routes can still be reflected between a client and a non-client.

Confederation

Confederation is another method to manage growing IBGP connections in an AS. It splits an AS into multiple sub-ASs. In each sub-AS, IBGP peers are fully meshed. As shown in Figure 9, intra-confederation EBGP connections are established between sub-ASs in AS 200.

Figure 9 Confederation network diagram



A non-confederation BGP speaker does not need to know sub-ASs in the confederation. To the BGP speaker, the confederation is one AS and the confederation ID is the AS number. In the above figure, AS 200 is the confederation ID.

Confederation has a deficiency. When you change an AS into a confederation, you must reconfigure the routers, and the topology will be changed.

In large-scale BGP networks, you can use both route reflector and confederation.

MP-BGP

Supported address families

BGP-4 can only advertise IPv4 unicast routing information. Multiprotocol Extensions for BGP-4 (MP-BGP) can advertise routing information for the following address families:

- IPv6 unicast address family.
- IPv4 multicast address family and IPv6 multicast address family.

  PIM uses static and dynamic unicast routes (including RIP, OSPF, IS-IS, BGP) to perform RPF check before creating multicast routing entries. When the multicast and unicast topologies are different, you can use MP-BGP to advertise the routes for RPF check. MP-BGP stores the routes in the BGP multicast routing table. For more information about PIM and RPF check, see *IP Multicast Configuration Guide*.
- VPNv4 address family and VPNv6 address family.

MP-BGP extended attributes

Prefixes and next hops are key routing information. BGP-4 uses UPDATE messages to carry the following information:

- Feasible route prefixes in the Network Layer Reachability Information (NLRI) field.
- Unfeasible route prefixes in the withdrawn routes field.
- Next hops in the NEXT_HOP attribute.

BGP-4 cannot carry routing information for multiple network layer protocols.

To support multiple network layer protocols, MP-BGP defines the following path attributes:

- **MP_REACH_NLRI**—Carries feasible route prefixes and next hops for multiple network layer protocols.
- **MP_UNREACH_NLRI**—Carries unfeasible route prefixes for multiple network layer protocols.

MP-BGP uses these two attributes to advertise feasible and unfeasible routes for different network layer protocols. BGP speakers not supporting MP-BGP ignore updates containing these attributes and do not forward them to its peers.

Address family

MP-BGP uses address families and subsequent address families to identify different network layer protocols for routes contained in the MP_REACH_NLRI and MP_UNREACH_NLRI attributes. For example, an Address Family Identifier (AFI) of 2 and a Subsequent Address Family Identifier (SAFI) of 1 identify IPv6 unicast routing information carried in the MP_REACH_NLRI attribute. For address family values, see RFC 1700.

BGP multi-instance

A BGP router can run multiple BGP processes. Each BGP process corresponds to a BGP instance. BGP maintains an independent routing table for each BGP instance.

BGP configuration views

BGP uses different views to manage routing information for different BGP instances, VPN instances, and address families. Most BGP commands are available in all BGP views. BGP supports multiple VPN instances by establishing a separate routing table for each VPN instance.

Table 1 describes different BGP configuration views.

Table 1 BGP configuration views

View names	Ways to enter the views	Remarks

BGP instance view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc]`
		You can create a BGP instance and enter its view by specifying the **`instance`** keyword in the **`bgp`** command.
		Configurations in this view apply to all public address families for the specified BGP instance. Some configurations (such as confederation, GR, and logging configurations) also apply to the address families of VPN instances.

BGP IPv4 unicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family ipv4 unicast`
	`[Sysname-bgp-abc-ipv4]`
		Configurations in this view apply to public IPv4 unicast routes and peers of the specified BGP instance.

BGP IPv6 unicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family ipv6 unicast`
	`[Sysname-bgp-abc-ipv6]`
		Configurations in this view apply to public IPv6 unicast routes and peers of the specified BGP instance.

BGP IPv4 multicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
		Configurations in this view apply to IPv4 multicast routes and peers of

View names	Ways to enter the views	Remarks

	`[Sysname-bgp-abc] address-family ipv4 multicast`
	`[Sysname-bgp-abc-mul-ipv4]`
		the specified BGP instance.

BGP IPv6 multicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family ipv6 multicast`
	`[Sysname-bgp-abc-mul-ipv6]`
		Configurations in this view apply to IPv6 multicast routes and peers of the specified BGP instance.

BGP VPNv4 address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family vpnv4`
	`[Sysname-bgp-abc-vpnv4]`
		Configurations in this view apply to VPNv4 routes and peers of the specified BGP instance.

BGP VPNv6 address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family vpnv6`
	`[Sysname-bgp-abc-vpnv6]`
		Configurations in this view apply to VPNv6 routes and peers of the specified BGP instance.

BGP-VPN instance view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] ip vpn-instance vpn1`
	`[Sysname-bgp-abc-vpn1]`
		Configurations in this view apply to all address families in the specified VPN instance of the specified BGP instance.

BGP-VPN IPv4 unicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] ip vpn-instance vpn1`
	`[Sysname-bgp-abc-vpn1] address-family ipv4 unicast`
	`[Sysname-bgp-abc-ipv4-vpn1]`
		Configurations in this view apply to IPv4 unicast routes and peers in the specified VPN instance of the specified BGP instance.

BGP-VPN IPv6 unicast address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] ip vpn-instance vpn1`
	`[Sysname-bgp-abc-vpn1] address-family ipv6 unicast`
	`[Sysname-bgp-abc-ipv6-vpn1]`
		Configurations in this view apply to IPv6 unicast routes and peers in the specified VPN instance of the specified BGP instance.

BGP IPv4 MVPN address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family ipv4 mvpn`
	`[Sysname-bgp-abc-mvpn]`
		Configurations in this view apply to MVPN routes and peers of the specified BGP instance.

BGP LS address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family link-state`
	`[Sysname-bgp-abc-ls]`
		Configurations in this view apply to LS messages and peers of the specified BGP instance.

View names	Ways to enter the views	Remarks

BGP IPv4 RT filter address family view	`<Sysname> system-view`
	`[Sysname] bgp 100 instance abc`
	`[Sysname-bgp-abc] address-family ipv4 rtfilter`
	`[Sysname-bgp-abc-rtf-ipv4]`
		Configurations in this view apply to IPv4 RT filter routes and peers of the specified BGP instance.

Protocols and standards

- RFC 1700, *ASSIGNED NUMBERS*
- RFC 1997, *BGP Communities Attribute*
- RFC 2439, *BGP Route Flap Damping*
- RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*
- RFC 2918, *Route Refresh Capability for BGP-4*
- RFC 3107, *Carrying Label Information in BGP-4*
- RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*
- RFC 4275, *BGP-4 MIB Implementation Survey*
- RFC 4277, *Experience with the BGP-4 Protocol*
- RFC 4360, *BGP Extended Communities Attribute*
- RFC 4451, *BGP MULTI_EXIT_DISC (MED) Consideration*
- RFC 4456, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP*
- RFC 4486, *Subcodes for BGP Cease Notification Message*
- RFC 4724, *Graceful Restart Mechanism for BGP*
- RFC 4760, *Multiprotocol Extensions for BGP-4*
- RFC 5004, *Avoid BGP Best Path Transitions from One External to Another*
- RFC 5065, *Autonomous System Confederations for BGP*
- RFC 5082, *The Generalized TTL Security Mechanism (GTSM)*
- RFC 5668, *4-Octet AS Specific BGP Extended Community*
- RFC 6608, *Subcodes for BGP Finite State Machine Error*
- RFC 6624, *Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling*
- RFC 6793, *BGP Support for Four-Octet Autonomous System (AS) Number Space*
- RFC 7752, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*

Building basic BGP networks

Restrictions and guidelines: BGP configuration

You can create multiple public address families for a BGP instance. However, each public address family (except for public VPNv4 and VPNv6 address families) can belong to only one BGP instance.

You can create multiple VPN instances for a BGP instance, and each VPN instance can have multiple address families. A VPN instance can belong to only one BGP instance.

You cannot specify the same peer for the same address family of different BGP instances.

Different BGP instances can have the same AS number but cannot have the same name.

Basic BGP network configuration tasks at a glance (IPv4 unicast/IPv4 multicast)

To build basic BGP networks for the IPv4 unicast or IPv4 multicast address family, perform the following tasks:

1. Configuring basic BGP
   a. Enabling BGP
   b. Configuring a BGP peer
   c. Configuring dynamic BGP peers
   d. Configuring an IBGP peer group

      Configure BGP peer groups on large-scale BGP networks for easy configuration and maintenance.
   e. Configuring an EBGP peer group

      Configure BGP peer groups on large-scale BGP networks for easy configuration and maintenance.
   f. (Optional.) Specifying the source address of TCP connections
2. Controlling BGP route generation

   Choose the following tasks as needed:
   - Injecting a local network
   - Redistributing IGP routes
   - (Optional.) Configuring BGP route summarization
   - (Optional.) Advertising a default route to a peer or peer group
3. (Optional.) Controlling BGP route advertisement
   - Advertising optimal routes in the IP routing table

     BGP cannot advertise optimal routes in the IP routing table for IPv4 multicast address family.
   - Configuring BGP route distribution filtering policies
   - Configuring BGP to delay sending route updates on reboot
4. (Optional.) Controlling BGP route reception
   - Limiting routes received from a peer or peer group
   - Configuring BGP route reception filtering policies
   - Configuring the SoO attribute

5. (Optional.) Configuring BGP timers
    - Configuring the keepalive interval and hold time
    - Setting the session retry timer
    - Configuring the interval for sending updates for the same route
6. (Optional.) Configuring BGP logging and notifications
    - Enabling logging for session state changes
    - Configuring BGP network management

Basic BGP network configuration tasks at a glance (IPv6 unicast/IPv6 multicast)

To build basic BGP networks for the IPv6 unicast or IPv6 multicast address family, perform the following tasks:

1. Configuring basic BGP
    a. Enabling BGP
    b. Configuring a BGP peer
    c. Configuring dynamic BGP peers
    d. Configuring an IBGP peer group

       Configure BGP peer groups on large-scale BGP networks for easy configuration and maintenance.
    e. Configuring an EBGP peer group

       Configure BGP peer groups on large-scale BGP networks for easy configuration and maintenance.
    f. (Optional.) Specifying the source address of TCP connections
2. Controlling BGP route generation

   Choose the following tasks as needed:
    - Injecting a local network
    - Redistributing IGP routes
    - (Optional.) Configuring BGP route summarization
    - (Optional.) Advertising a default route to a peer or peer group
3. (Optional.) Controlling BGP route advertisement
    - Advertising optimal routes in the IP routing table

      BGP cannot advertise optimal routes in the IP routing table for IPv6 multicast address family.
    - Configuring BGP route distribution filtering policies
    - Configuring BGP to delay sending route updates on reboot
4. (Optional.) Controlling BGP route reception
    - Limiting routes received from a peer or peer group
    - Configuring BGP route reception filtering policies
    - Configuring the SoO attribute
5. (Optional.) Configuring BGP timers
    - Configuring the keepalive interval and hold time
    - Setting the session retry timer
    - Configuring the interval for sending updates for the same route

6. (Optional.) Configuring BGP logging and notifications
   - Enabling logging for session state changes
   - Configuring BGP network management

Configuring basic BGP

Enabling BGP

Restrictions and guidelines

A router ID is the unique identifier of a BGP router in an AS.

- To ensure the uniqueness of a router ID and enhance availability, specify in BGP instance view the IP address of a local loopback interface as the router ID. Different BGP instances can have the same router ID.
- If no router ID is specified in BGP instance view, the global router ID is used.
- To modify a non-zero router ID of a BGP instance , use the **router-id** command in BGP instance view, rather than the **router id** command in system view.
- If you specify a router ID in BGP instance view and then remove the interface that owns the router ID, the router does not select a new router ID. To select a new router ID, use the **undo router-id** command in BGP instance view.

Procedure

1. Enter system view.

   system-view

2. Configure a global router ID.

   router id *router-id*

   By default, no global router ID is configured.

   If no global router ID is configured, the following rules apply:

   - If loopback interfaces configured with an IP address exist, BGP uses the highest loopback interface IP address as the router ID.
   - If no loopback interface IP address is available, BGP uses the highest physical interface IP address as the route ID regardless of the interface status.

3. Enable BGP and enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

   By default, BGP is disabled and no BGP instances exist.

4. (Optional.) Configure a router ID for the BGP instance.

   router-id *router-id*

   By default, no router ID is configured for a BGP instance, and the BGP instance uses the global router ID configured by the **router-id** command in system view.

5. (Optional.) Enter BGP-VPN instance view.

   ip vpn-instance *vpn-instance-name*

   The specified VPN instance must have been created and have an RD.

6. (Optional.) Configure a router ID for the BGP VPN instance.

   router-id { *router-id* | **auto-select** }

   By default, no router ID is configured for a BGP VPN instance.

The BGP VPN instance uses the router ID configured in BGP instance view. If no router ID is configured in BGP instance view, the BGP VPN instance uses the global router ID configured in system view.

Configuring a BGP peer

Restrictions and guidelines

A BGP peer at an IPv6 link-local address must be directly connected to the local router. On the local router, you must use the **peer connect-interface** command to specify the interface directly connected to the BGP peer as the source interface of TCP connections.

Procedure (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an IPv4 BGP peer and specify its AS number.

   peer *ipv4-address* **as-number** *as-number*

4. (Optional.) Configure a description for a peer.

   peer *ipv4-address* **description** *text*

   By default, no description is configured for a peer.

5. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

6. Enable the router to exchange IPv4 unicast routing information with the specified peer.

   peer *ipv4-address* **enable**

   By default, the router cannot exchange IPv4 unicast routing information with the peer.

Procedure (IPv6 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an IPv6 BGP peer and specify its AS number.

   peer *ipv6-address* **as-number** *as-number*

4. (Optional.) Configure a description for a peer.

   peer *ipv6-address* **description** *text*

   By default, no description is configured for a peer.

5. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]

6. Enable the router to exchange IPv6 unicast routing information with the specified peer.

   peer *ipv6-address* **enable**

   By default, the router cannot exchange IPv6 unicast routing information with the peer.

Procedure (IPv4 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an IPv4 BGP peer and specify its AS number.

   peer *ipv4-address* **as-number** *as-number*

4. (Optional.) Configure a description for the peer.

   peer *ipv4-address* **description** *text*

   By default, no description is configured for a peer.

5. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

6. Enable the router to exchange IPv4 unicast routing information used for RPF check with the specified peer.

   peer *ipv4-address* **enable**

   By default, the router cannot exchange IPv4 unicast routing information used for RPF check with the peer.

Procedure (IPv6 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an IPv6 BGP peer and specify its AS number.

   peer *ipv6-address* **as-number** *as-number*

4. (Optional.) Configure a description for the peer.

   peer *ipv6-address* **description** *text*

   By default, no description is configured for a peer.

5. Create the BGP IPv6 multicast address family and enter its view.

   address-family ipv6 multicast

6. Enable the router to exchange IPv6 unicast routing information used for RPF check with the specified peer.

   peer *ipv6-address* **enable**

   By default, the router cannot exchange IPv6 unicast routing information used for RPF check with the peer.

Configuring dynamic BGP peers

About this task

This feature enables BGP to establish dynamic BGP peer relationships with devices in a network. BGP accepts connection requests from the network but it does not initiate connection requests to the network.

After a device in the network initiates a connection request, BGP establishes a dynamic peer relationship with the device.

If multiple BGP peers reside in the same network, you can use this feature to simplify BGP peer configuration.

Restrictions and guidelines

For a remote device to establish a peer relationship with the local device, you must specify the IP address of the local device on the remote device.

A BGP peer at an IPv6 link-local address must be directly connected to the local router. On the local router, you must use the **peer connect-interface** command to specify the interface directly connected to the BGP peer as the source interface of TCP connections.

Procedure (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Specify devices in a network as dynamic BGP peers and specify an AS number for the peers.

   peer *ipv4-address mask-length* **as-number** *as-number*

4. (Optional.) Configure a description for dynamic BGP peers.

   peer *ipv4-address mask-length* **description** *text*

   By default, no description is configured for dynamic BGP peers.

5. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

6. Enable BGP to exchange IPv4 unicast routing information with dynamic BGP peers in the specified network.

   peer *ipv4-address mask-length* **enable**

   By default, BGP cannot exchange IPv4 unicast routing information with dynamic BGP peers.

Procedure (IPv6 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

bgp *as-number* [**instance** *instance-name*]

ip vpn-instance *vpn-instance-name*

3. Specify devices in a network as dynamic BGP peers and specify an AS number for the peers.

   peer *ipv6-address prefix-length* **as-number** *as-number*

4. (Optional.) Configure a description for dynamic BGP peers.

   peer *ipv6-address prefix-length* **description** *text*

   By default, no description is configured for dynamic BGP peers.

5. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]

6. Enable BGP to exchange IPv6 unicast routing information with dynamic BGP peers in the specified network.

   peer *ipv6-address prefix-length* **enable**

   By default, BGP cannot exchange IPv6 unicast routing information with dynamic BGP peers.

Procedure (IPv4 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Specify devices in a network as dynamic BGP peers and specify an AS number for the peers.

   peer *ipv4-address mask-length* **as-number** *as-number*

4. (Optional.) Configure a description for dynamic BGP peers.

   peer *ipv4-address mask-length* **description** *text*

   By default, no description is configured for dynamic BGP peers.

5. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

6. Enable BGP to exchange IPv4 unicast routing information used for RPF check with dynamic BGP peers in the specified network.

   peer *ipv4-address mask-length* **enable**

   By default, BGP cannot exchange IPv4 unicast routing information used for RPF check with dynamic BGP peers.

Procedure (IPv6 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Specify devices in a network as dynamic BGP peers and specify an AS number for the peers.

   peer *ipv6-address prefix-length* **as-number** *as-number*

4. (Optional.) Configure a description for dynamic BGP peers.

   peer *ipv6-address prefix-length* **description** *text*

   By default, no description is configured for dynamic BGP peers.

5. Create the BGP IPv6 multicast address family and enter its view.

   address-family ipv6 multicast

6. Enable BGP to exchange IPv6 unicast routing information used for RPF check with dynamic BGP peers in the specified network.

   peer *ipv6-address prefix-length* **enable**

   By default, BGP cannot exchange IPv6 unicast routing information used for RPF check with dynamic BGP peers.

Configuring an IBGP peer group

About this task

A peer group is an IBGP peer group if peers in it belong to the same AS as the local router.

After you create an IBGP peer group and then add a peer into it, the system creates the peer in BGP instance view and specifies the local AS number for the peer.

Restrictions and guidelines

A BGP peer at an IPv6 link-local address must be directly connected to the local router. On the local router, you must use the **peer connect-interface** command to specify the interface directly connected to the BGP peer as the source interface of TCP connections.

If you configure a BGP setting at both the peer group and the peer level, the most recent configuration takes effect on the peer.

Procedure (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an IBGP peer group.

   group *group-name* [**internal**]

4. Add a peer into the IBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the local AS number.

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

7. Enable the router to exchange IPv4 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information with the peers.

Procedure (IPv4 multicast address family)

1. Enter system view.

system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an IBGP peer group.

   group *group-name* [**internal**]

4. Add an IPv4 peer into the IBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the local AS number.

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

7. Enable the router to exchange IPv4 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information used for RPF check with the peers in the peer group.

Procedure (IPv6 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an IBGP peer group.

   group *group-name* [**internal**]

4. Add a peer into the IBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the local AS number.

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]

7. Enable the router to exchange IPv6 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information with the peers.

Procedure (IPv6 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an IBGP peer group.

   group *group-name* [**internal**]

4. Add a peer into the IBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the local AS number.

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv6 multicast address family and enter its view.

   address-family ipv6 multicast

7. Enable the router to exchange IPv6 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information used for RPF check with the peers in the peer group.

Configuring an EBGP peer group

About this task

A peer group is an EBGP peer group if peers in it belong to different ASs.

If peers in an EBGP group belong to the same external AS, the EBGP peer group is a pure EBGP peer group. If not, it is a mixed EBGP peer group.

Restrictions and guidelines

Use one of the following methods to configure an EBGP peer group:

- **Method 1**—Create an EBGP peer group, specify its AS number, and add peers into it. All the added peers have the same AS number. All peers in the peer group have the same AS number as the peer group. You can specify an AS number for a peer before adding it into the peer group. The AS number must be the same as that of the peer group.
- **Method 2**—Create an EBGP peer group, specify an AS number for a peer, and add the peer into the peer group. Peers added in the group can have different AS numbers.
- **Method 3**—Create an EBGP peer group and add a peer with an AS number into it. Peers added in the group can have different AS numbers.

If you configure a BGP setting at both the peer group and the peer level, the most recent configuration takes effect on the peer.

Configuring an EBGP peer group by using Method 1 (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

bgp *as-number* [**instance** *instance-name*]

- Execute the following commands in sequence to enter BGP-VPN instance view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

3. Create an EBGP peer group.

   group *group-name* **external**

4. Specify the AS number of the group.

   peer *group-name* **as-number** *as-number*

   By default, no AS number is specified.

   If a peer group contains peers, you cannot remove or change its AS number.

5. Add a peer into the EBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *group-name* **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

7. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

8. Enable the router to exchange IPv4 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 2 (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an EBGP peer group.

   group *group-name* **external**

4. Create an IPv4 BGP peer and specify its AS number.

   peer *ipv4-address* [*mask-length*] **as-number** *as-number*

5. Add the peer into the EBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *ipv4-address* [*mask-length*] **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

peer *group-name* **description** *text*

By default, no description is configured for the peer group.

7. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

8. Enable the router to exchange IPv4 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 3 (IPv4 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an EBGP peer group.

   group *group-name* **external**

4. Add a peer into the EBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* **as-number** *as-number*

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv4 unicast address family or BGP-VPN IPv4 unicast address family and enter its view.

   address-family ipv4 [**unicast**]

7. Enable the router to exchange IPv4 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 1 (IPv4 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an EBGP peer group.

   group *group-name* **external**

4. Specify the AS number of the group.

   peer *group-name* **as-number** *as-number*

   By default, no AS number is specified.

   If a peer group contains peers, you cannot remove or change its AS number.

5. Add an IPv4 BGP peer into the EBGP peer group.

peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

The **as-number** *as-number* option must specify the same AS number as the **peer** *group-name* **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

7. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

8. Enable the router to exchange IPv4 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information used for RPF check with the peers in the group.

Configuring an EBGP peer group by using Method 2 (IPv4 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an EBGP peer group.

   group *group-name* **external**

4. Create an IPv4 BGP peer and specify its AS number.

   peer *ipv4-address* [*mask-length*] **as-number** *as-number*

5. Add the peer into the EBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *ipv4-address* [*mask-length*] **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

7. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

8. Enable the router to exchange IPv4 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information used for RPF check with the peers in the group.

Configuring an EBGP peer group by using Method 3 (IPv4 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an EBGP peer group.

   group *group-name* **external**

4. Add an IPv4 BGP peer into the EBGP peer group.

   peer *ipv4-address* [*mask-length*] **group** *group-name* **as-number** *as-number*

5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv4 multicast address family and enter its view.

   address-family ipv4 multicast

7. Enable the router to exchange IPv4 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv4 unicast routing information used for RPF check with the peers.

Configuring an EBGP peer group by using Method 1 (IPv6 unicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Create an EBGP peer group.

   group *group-name* **external**

4. Specify the AS number of the group.

   peer *group-name* **as-number** *as-number*

   By default, no AS number is specified.

   If a peer group contains peers, you cannot remove or change its AS number.

5. Add a peer into the EBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *group-name* **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

7. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]

8. Enable the router to exchange IPv6 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 2 (IPv6 unicast address family)

1. Enter system view.

system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Create an EBGP peer group.

   group *group-name* **external**
4. Create an IPv6 BGP peer and specify its AS number.

   peer *ipv6-address* [*prefix-length*] **as-number** *as-number*
5. Add the peer into the EBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *ipv6-address* [*prefix-length*] **as-number** *as-number* command.
6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.
7. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]
8. Enable the router to exchange IPv6 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 3 (IPv6 unicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Create an EBGP peer group.

   group *group-name* **external**
4. Add a peer into the EBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* **as-number** *as-number*
5. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

6. Create the BGP IPv6 unicast address family or BGP-VPN IPv6 unicast address family and enter its view.

   address-family ipv6 [**unicast**]

7. Enable the router to exchange IPv6 unicast routing information with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information with the peers.

Configuring an EBGP peer group by using Method 1 (IPv6 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an EBGP peer group.

   group *group-name* **external**

4. Specify the AS number of the group.

   peer *group-name* **as-number** *as-number*

   By default, no AS number is specified.

   If a peer group contains peers, you cannot remove or change its AS number.

5. Add an IPv6 BGP peer into the EBGP peer group.

   peer *ipv6-address* [*prefix-length*] **group** *group-name* [**as-number** *as-number*]

   The **as-number** *as-number* option must specify the same AS number as the **peer** *group-name* **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   peer *group-name* **description** *text*

   By default, no description is configured for the peer group.

7. Create the BGP IPv6 multicast address family and enter its view.

   address-family ipv6 multicast

8. Enable the router to exchange IPv6 unicast routing information used for RPF check with peers in the specified peer group.

   peer *group-name* **enable**

   By default, the router cannot exchange IPv6 unicast routing information used for RPF check with the peers in the group.

Configuring an EBGP peer group by using Method 2 (IPv6 multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Create an EBGP peer group.

   group *group-name* **external**

4. Create an IPv6 BGP peer and specify its AS number.

   peer *ipv6-address* [*prefix-length*] **as-number** *as-number*

5. Add the peer into the EBGP peer group.

```
peer ipv6-address [ prefix-length ] group group-name [ as-number
as-number ]
```

The **as-number** *as-number* option must specify the same AS number as the **peer** *ipv6-address* [*prefix-length*] **as-number** *as-number* command.

6. (Optional.) Configure a description for the peer group.

   ```
   peer group-name description text
   ```

   By default, no description is configured for the peer group.

7. Create the BGP IPv6 multicast address family and enter its view.

   ```
   address-family ipv6 multicast
   ```

8. Enable the router to exchange IPv6 unicast routing information used for RPF check with peers in the specified peer group.

   ```
   peer group-name enable
   ```

   By default, the router cannot exchange IPv6 unicast routing information used for RPF check with the peers in the group.

Configuring an EBGP peer group by using Method 3 (IPv6 multicast address family)

1. Enter system view.

   ```
   system-view
   ```

2. Enter BGP instance view.

   ```
   bgp as-number [ instance instance-name ]
   ```

3. Create an EBGP peer group.

   ```
   group group-name external
   ```

4. Add an IPv6 BGP peer into the EBGP peer group.

   ```
   peer ipv6-address [ prefix-length ] group group-name as-number
   as-number
   ```

5. (Optional.) Configure a description for the peer group.

   ```
   peer group-name description text
   ```

   By default, no description is configured for the peer group.

6. Create the BGP IPv6 multicast address family and enter its view.

   ```
   address-family ipv6 multicast
   ```

7. Enable the router to exchange IPv6 unicast routing information used for RPF check with peers in the specified peer group.

   ```
   peer group-name enable
   ```

   By default, the router cannot exchange IPv6 unicast routing information used for RPF check with the peers in the group.

Specifying the source address of TCP connections

About this task

BGP uses TCP as the transport layer protocol. Perform this task in the following scenarios to specify the source address or source interface of TCP connections to a peer or peer group:

- The peer's IPv4/IPv6 address does not belong to the interface directly connected to the local router. To ensure successful TCP connection establishment, use one of the following methods:
  - Specify the interface to which the IPv4/IPv6 address belongs as the source interface on the peer.
  - Specify the IPv4/IPv6 address of the interface directly connected to the local router as the source address on the peer.

- A BGP peer at an IPv6 link-local address must be directly connected to the local router. On the local router, you must use the **peer connect-interface** command to specify the interface directly connected to the BGP peer as the source interface of TCP connections.
- On a BGP router that has multiple links to a peer, the source interface for TCP connection changes because the primary source interface fails. To avoid this problem, specify a loopback interface as the source interface or specify the IP address of a loopback interface as the source address.
- You want to establish multiple BGP sessions to a router. In this case, BGP might fail to determine the source address for each TCP connection based on the optimal route to the peer. To prevent this problem, use one of the following methods:
    - If the BGP sessions use IP addresses of different interfaces, specify a source interface or source address for each session.
    - If the BGP sessions use different IP addresses of the same interface, specify a source address for each session.

Restrictions and guidelines

BGP immediately tears down the session to an IBGP peer or peer group when the following conditions exist:

- The source interface of TCP connections to the IBGP peer or peer group is a physical interface.
- The source interface fails and the link to the IBGP peer or peer group goes down.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
    - Enter BGP instance view.

      bgp *as-number* [**instance** *instance-name*]

    - Execute the following commands in sequence to enter BGP-VPN instance view:

      bgp *as-number* [**instance** *instance-name*]

      ip vpn-instance *vpn-instance-name*

3. Specify the source address or source interface of TCP connections to a peer or peer group.
    - Specify the source address of TCP connections to a peer or peer group.

      peer { *group-name* | *ipv4-address* [*mask-length*] } **source-address** *source-ipv4-address*

    - Specify the source interface of TCP connections to a peer or peer group.

      peer { *group-name* | *ipv4-address* [*mask-length*] } **connect-interface** *interface-type interface-number*

   By default, BGP uses the primary IPv4 address of the output interface in the optimal route to a peer or peer group as the source address of TCP connections to the peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
    - Enter BGP instance view.

      bgp *as-number* [**instance** *instance-name*]

    - Execute the following commands in sequence to enter BGP-VPN instance view:

      bgp *as-number* [**instance** *instance-name*]

      ip vpn-instance *vpn-instance-name*

3. Specify the source IPv6 address or source interface of TCP connections to a peer or peer group.
   - Specify the source IPv6 address of TCP connections to a peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **source-address** *source-ipv6-address*
   - Specify the source interface of TCP connections to a peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **connect-interface** *interface-type interface-number*

   By default, BGP uses the IPv6 address of the output interface in the optimal route to the BGP peer or peer group as the source address of TCP connections to the peer or peer group.

Controlling BGP route generation

Injecting a local network

About this task

Perform this task to inject a network in the local routing table to the BGP routing table, so BGP can advertise the network to BGP peers. The ORIGIN attribute of BGP routes advertised in this way is IGP. You can also use a routing policy to control route advertisement.

The specified network must be available and active in the local IP routing table.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast
3. Configure BGP to advertise a local network.

   network *ipv4-address* [*mask-length* | *mask*] [**route-policy** *route-policy-name*]

   By default, BGP does not advertise local networks.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast
3. Configure BGP to advertise a local network.

   network *ipv6-address prefix-length* [**route-policy** *route-policy-name*]

   By default, BGP does not advertise local networks.

Redistributing IGP routes

About this task

Perform this task to configure route redistribution from an IGP to BGP.

By default, BGP does not redistribute default IGP routes. You can use the **default-route imported** command to redistribute default IGP routes into the BGP routing table.

The ORIGIN attribute of BGP routes redistributed from IGPs is INCOMPLETE.

Only active routes can be redistributed. To view route state information, use the **display ip routing-table protocol** or **display ipv6 routing-table protocol** command. For more information about the commands, see *Layer 3—IP Routing Command Reference*.

If you execute the **import-route** command multiple times for an IGP process, the most recent configuration takes effect. To redistribute more routes from an IGP process without overwriting the routes redistributed before, use the **import-route-append** command.

When you execute both the **import-route** and **import-route-append** commands for an IGP process, the commands take effect as follows:

- A route is redistributed as long as it matches the criteria of either command.
- If a route matches the criteria of both commands, the route is redistributed, and the apply clauses in the routing policies specified in the two commands take effect as follows:
  - If the apply clauses do not conflict, all apply clauses take effect.
  - If conflicts occur between the apply clauses, only the apply clauses in the **import-route-append** command take effect.
- The MED value specified by the **import-route-append** command takes precedence over that specified by the **import-route** command.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast
3. Enable route redistribution from the specified IGP into BGP.
   - Redistribute **IS-IS, OSPF, or RIP routes.**

     import-route { **isis** | **ospf** | **rip** } [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]
   - Redistribute direct or static routes.

     import-route { **direct** | **static** } [**med** *med-value* | **route-policy** *route-policy-name*] *

   By default, BGP does not redistribute IGP routes.
4. (Optional.) Redistribute routes from an IGP without overwriting the routes redistributed by the **import-route** command.
   - Redistribute IS-IS, OSPF, or RIP routes.

     import-route-append { **isis** | **ospf** | **rip** } [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]
   - Redistribute direct or static routes.

     import-route-append { **direct** | **static** } [**med** *med-value* | **route-policy** *route-policy-name*] *

   By default, BGP does not redistribute IGP routes.
5. (Optional.) Enable default route redistribution into BGP.

   default-route imported

   By default, BGP does not redistribute default routes.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv6 multicast

3. Enable route redistribution from the specified IGP into BGP.

   - Redistribute IPv6 **IS-IS, OSPFv3, or RIPng routes.**

     import-route { **isisv6** | **ospfv3** | **ripng** } [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]

   - Redistribute direct or static routes.

     import-route { **direct** | **static** } [**med** *med-value* | **route-policy** *route-policy-name*] *

   By default, BGP does not redistribute IGP routes.

4. (Optional.) Redistribute routes from an IGP without overwriting the routes redistributed by the **import-route** command.

   - Redistribute IPv6 IS-IS, OSPFv3, or RIPng routes.

     import-route-append { **isisv6** | **ospfv3** | **ripng** } [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]

   - Redistribute direct or static routes.

     import-route-append { **direct** | **static** } [**med** *med-value* | **route-policy** *route-policy-name*] *

   By default, BGP does not redistribute IGP routes.

5. (Optional.) Enable default route redistribution into BGP.

   default-route imported

   By default, BGP does not redistribute default routes.

Configuring BGP route summarization

About this task

Route summarization can reduce the number of redistributed routes and the routing table size. IPv4 BGP supports automatic route summarization and manual route summarization. Manual summarization takes precedence over automatic summarization. IPv6 BGP supports only manual route summarization.

Automatic route summarization enables BGP to summarize IGP subnet routes redistributed by the **import-route** command, so BGP advertises only natural network routes.

By configuring manual route summarization, you can do the following:

- Summarize both redistributed routes and routes injected using the **network** command.
- Determine the mask length for a summary route.

Restrictions and guidelines for configuring BGP route summarization

The output interface of a BGP summary route is Null 0 on the originating router. Therefore, a summary route must not be an optimal route on the originating router. Otherwise, BGP will fail to forward packets matching the route. If a summarized specific route has the same mask as the summary route, but has a lower priority, the summary route becomes the optimal route. To ensure correct packet forwarding, change the priority of the summary or specific route to make the specific route the optimal route.

Configuring automatic route summarization (IPv4 unicast/multicast address family)

1. Enter system view.

   `system-view`

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure automatic route summarization.

   `summary automatic`

   By default, automatic route summarization is not configured.

Configuring manual route summarization (IPv4 unicast/multicast address family)

1. Enter system view.

   `system-view`

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Create a summary route in the BGP routing table.

   aggregate *ipv4-address* { *mask-length* | *mask* } [**as-set** | **attribute-policy** *route-policy-name* | **detail-suppressed** | **origin-policy** *route-policy-name* | **suppress-policy** *route-policy-name*] *

   By default, no summary routes are configured.

Configuring BGP manual route summarization (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Create a summary route in the IPv6 BGP routing table.

   aggregate *ipv6-address prefix-length* [**as-set** | **attribute-policy** *route-policy-name* | **detail-suppressed** | **origin-policy** *route-policy-name* | **suppress-policy** *route-policy-name*] *

   By default, no summary routes are configured.

Advertising a default route to a peer or peer group

About this task

Perform this task to advertise a default BGP route with the next hop being the advertising router to a peer or peer group.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

bgp *as-number* [**instance** *instance-name*]

ip vpn-instance *vpn-instance-name*

address-family ipv4 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 multicast

3. Advertise a default route to a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **default-route-advertise** [**route-policy** *route-policy-name*]

   By default, no default route is advertised.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Advertise a default route to a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **default-route-advertise** [**route-policy** *route-policy-name*]

   By default, no default route is advertised.

Controlling BGP route advertisement

Advertising optimal routes in the IP routing table

About this task

By default, BGP advertises optimal routes in the BGP routing table, which may not be optimal in the IP routing table. This task allows you to advertise BGP routes that are optimal in the IP routing table.

Procedure (IPv4 unicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

bgp *as-number* [**instance** *instance-name*]

3. Enable BGP to advertise optimal routes in the IP routing table.

   advertise-rib-active

   By default, BGP advertises optimal routes in the BGP routing table.

4. Enter BGP IPv4 unicast address family view or BGP-VPN IPv4 unicast address family view.
   - Enter BGP IPv4 unicast address family view.

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

5. Enable BGP to advertise optimal routes in the IP routing table of the address family in the VPN instance.

   advertise-rib-active

   By default, the setting is the same as that in BGP instance view.

Procedure (IPv6 unicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable BGP to advertise optimal routes in the IPv6 routing table.

   advertise-rib-active

   By default, BGP advertises optimal routes in the BGP routing table.

4. Enter BGP IPv6 unicast address family view or BGP-VPN IPv6 unicast address family view.
   - Enter BGP IPv6 unicast address family view.

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast]**

5. Enable BGP to advertise optimal routes in the IPv6 routing table of the address family in the VPN instance.

   advertise-rib-active

   By default, the setting is the same as that in BGP instance view.

Configuring BGP route distribution filtering policies

About this task

To configure BGP route distribution filtering policies, use the following methods:

- Use an ACL or prefix list to filter routing information advertised to all peers.
- Use a routing policy, ACL, conditional advertisement policy (existent policy or nonexistent policy), AS path list, or prefix list to filter routing information advertised to a peer or peer group.

If you configure multiple filtering policies, apply them in the following sequence:

1. **peer prefix-list export**

2. **`peer filter-policy export`**
3. **`peer as-path-acl export`**
4. **`filter-policy export`**
5. **`peer advertise-policy exist-policy`**
6. **`peer advertise-policy non-exist-policy`**
7. **`peer route-policy export`**

Only routes passing all the configured policies can be advertised.

Prerequisites

Before you configure BGP routing filtering policies, configure the following filters used for route filtering as needed:

- ACL (see *ACL and QoS Configuration Guide*).
- Prefix list (see "Configuring routing policies").
- Routing policy (see "Configuring routing policies").
- AS path list (see "Configuring routing policies").

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure BGP route distribution filtering policies. Choose the options to configure as needed:
   - Reference an ACL or IP prefix list to filter advertised BGP routes.

     filter-policy { *ipv4-acl-number* | **prefix-list** *ipv4-prefix-list-name* } **export** [**direct** | { **isis** | **ospf** | **rip** } *process-id* | **static**]
   - Specify a routing policy as the existent policy to control route advertisement.

     peer { *group-name* | *ipv4-address* [*mask-length*] } **advertise-policy** *advertise-policy-name* **exist-policy** *exist-policy-name*

     This command is supported only in BGP IPv4 unicast address family view.
   - Specify a routing policy as the nonexistent policy to control route advertisement.

     peer { *group-name* | *ipv4-address* [*mask-length*] } **advertise-policy** *advertise-policy-name* **non-exist-policy** *non-exist-policy-name*

  This command is supported only in BGP IPv4 unicast address family view.

- Reference a routing policy to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv4-address* [*mask-length*] } **route-policy** *route-policy-name* **export**

- Reference an ACL to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv4-address* [*mask-length*] } **filter-policy** { *ipv4-acl-number* } **export**

- Reference an AS path list to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv4-address* [*mask-length*] } **as-path-acl** *as-path-acl-number* **export**

- Reference an IPv4 prefix list to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv4-address* [*mask-length*] } **prefix-list** *ipv4-prefix-list-name* **export**

By default, no BGP distribution filtering policy is configured.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Configure BGP route distribution filtering policies. Choose the options to configure as needed:

   - Reference an ACL or IPv6 prefix list to filter advertised BGP routes.

     filter-policy { *ipv6-acl-number* | **prefix-list** *ipv6-prefix-list-name* } **export** [**direct** | { **isisv6** | **ospfv3** | **ripng** } *process-id* | **static**]

   - Specify a routing policy as the existent policy to control route advertisement.

     peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **advertise-policy** *advertise-policy-name* **exist-policy** *exist-policy-name*

     This command is supported only in BGP IPv6 unicast address family view.

   - Specify a routing policy as the nonexistent policy to control route advertisement.

     peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **advertise-policy** *advertise-policy-name* **non-exist-policy** *non-exist-policy-name*

This command is supported only in BGP IPv6 unicast address family view.

- Reference a routing policy to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **route-policy** *route-policy-name* **export**

- Reference an ACL to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **filter-policy** { *ipv6-acl-number* } **export**

- Reference an AS path list to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **as-path-acl** *as-path-acl-number* **export**

- Reference an IPv6 prefix list to filter BGP routes advertised to a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **prefix-list** *ipv6-prefix-list-name* **export**

By default, no BGP distribution filtering policy is configured.

Configuring BGP to delay sending route updates on reboot

About this task

This task reduces traffic loss. With this task performed, BGP delays sending route updates when it restores after a device reboot. During the delay time, BGP learns all routes from other neighbors, and then selects the optimal route. After the delay time elapses, BGP will advertise the optimal route.

You can specify a prefix list and configure BGP to immediately send route updates for routes that match the prefix list.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Configure BGP to delay sending route updates when it restores after a device reboot.

   bgp update-delay on-startup *seconds*

   By default, BGP immediately sends route updates to BGP peers in established state when it restores after a device reboot.

4. (Optional.) Configure BGP to immediately send route updates for routes that match a prefix list.

   bgp update-delay on-startup prefix-list *ipv4-prefix-list-name*

   By default, no prefix list is specified to filter routes.

Controlling BGP route reception

Limiting routes received from a peer or peer group

About this task

This feature can prevent attacks that send a large number of BGP routes to the router.

If the number of routes received from a peer or peer group exceeds the upper limit, the router takes one of the following actions based on your configuration:

- Tears down the BGP session to the peer or peer group and does not attempt to re-establish the session.
- Continues to receive routes from the peer or peer group and generates a log message.
- Retains the session to the peer or peer group, but it discards excess routes and generates a log message.
- Tears down the BGP session to the peer or peer group and, after a specific period of time, re-establishes a BGP session to the peer or peer group.

You can specify a percentage threshold for the router to generate a log message. When the ratio of the number of received routes to the maximum number reaches the percentage value, the router generates a log message.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv4 [ unicast ]
     ```

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     address-family ipv4 [ unicast ]
     ```

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv4 multicast
     ```

3. Specify the maximum number of routes that a router can receive from a peer or peer group.

   ```
   peer { group-name | ipv4-address [ mask-length ] } route-limit
   prefix-number [ { alert-only | discard | reconnect reconnect-time } |
   percentage-value ] *
   ```

   By default, the number of routes that a router can receive from a peer or peer group is not limited.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv6 [ unicast ]
     ```

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     ```

ip vpn-instance *vpn-instance-name*

address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv6 multicast

3. Specify the maximum number of routes that a router can receive from a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **route-limit** *prefix-number* [{ **alert-only** | **discard** | **reconnect** *reconnect-time* } | *percentage-value*] *

   By default, the number of routes that a router can receive from a peer or peer group is not limited.

Configuring BGP route reception filtering policies

About this task

You can use the following methods to configure BGP route reception filtering policies:

- Use an ACL or prefix list to filter routing information received from all peers.
- Use a routing policy, ACL, AS path list, or prefix list to filter routing information received from a peer or peer group.

If you configure multiple filtering policies, apply them in the following sequence:

1. **peer filter-policy import**
2. **peer prefix-list import**
3. **peer as-path-acl import**
4. **filter-policy import**
5. **peer route-policy import**

Only routes passing all the configured policies can be received.

Prerequisites

Before you configure BGP route reception filtering policies, configure the following filters used for route filtering as needed:

- ACL (see *ACL and QoS Configuration Guide*).
- Prefix list (see "Configuring routing policies").
- Routing policy (see "Configuring routing policies").
- AS path list (see "Configuring routing policies").

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

```
bgp as-number [ instance instance-name ]
ip vpn-instance vpn-instance-name
address-family ipv4 [ unicast ]
```

- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  ```
  bgp as-number [ instance instance-name ]
  address-family ipv4 multicast
  ```

3. Configure BGP route reception filtering policies. Choose the options to configure as needed:
   - Reference an ACL or IP prefix list to filter BGP routes received from all peers.

     ```
     filter-policy { ipv4-acl-number | prefix-list
     ipv4-prefix-list-name } import
     ```
   - Reference a routing policy to filter BGP routes received from a peer or peer group.

     ```
     peer { group-name | ipv4-address [ mask-length ] } route-policy
     route-policy-name import
     ```
   - Reference an ACL to filter BGP routes received from a peer or peer group.

     ```
     peer { group-name | ipv4-address [ mask-length ] } filter-policy
     { ipv4-acl-number } import
     ```
   - Reference an AS path list to filter BGP routes received from a peer or peer group.

     ```
     peer { group-name | ipv4-address [ mask-length ] } as-path-acl
     as-path-acl-number import
     ```
   - Reference an IPv4 prefix list to filter BGP routes received from a peer or peer group.

     ```
     peer { group-name | ipv4-address [ mask-length ] } prefix-list
     ipv4-prefix-list-name import
     ```

   By default, no route reception filtering is configured.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv6 [ unicast ]
     ```
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     address-family ipv6 [ unicast ]
     ```
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv6 multicast
     ```
3. Configure BGP route reception filtering policies. Choose the options to configure as needed:
   - Reference ACL or IPv6 prefix list to filter BGP routes received from all peers.

     ```
     filter-policy { ipv6-acl-number | prefix-list
     ipv6-prefix-list-name } import
     ```

- Reference a routing policy to filter BGP routes received from a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **route-policy** *route-policy-name* **import**

- Reference an ACL to filter BGP routes received from a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **filter-policy** { *ipv6-acl-number* } **import**

- Reference an AS path list to filter BGP routes received from a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **as-path-acl** *as-path-acl-number* **import**

- Reference an IPv6 prefix list to filter BGP routes received from a peer or peer group.

  peer { *group-name* | *ipv6-address* [*prefix-length*] } **prefix-list** *ipv6-prefix-list-name* **import**

By default, no route reception filtering is configured.

Configuring the SoO attribute

About this task

After you configure the SoO attribute for a BGP peer or peer group, BGP adds the SoO attribute into the route updates received from the BGP peer or peer group. In addition, before advertising route updates to the peer or peer group, BGP checks the SoO attribute of the route update against the configured SoO attribute. If they are the same, BGP does not advertise the route updates to the BGP peer or peer group.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure the SoO attribute for a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **soo** *site-of-origin*

   By default, no SoO attribute is configured for a peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast
3. Configure the SoO attribute for a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **soo** *site-of-origin*

   By default, no SoO attribute is configured for a peer or peer group.

Configuring BGP timers

Configuring the keepalive interval and hold time

About this task

BGP sends KEEPALIVE messages regularly to keep the BGP session between two routers.

If a router receives no KEEPALIVE or UPDATE message from a peer within the hold time, it tears down the session.

You can configure the keepalive interval and hold time globally or for a peer or peer group. The individual settings take precedence over the global settings.

The actual keepalive interval and hold time are determined as follows:

- If the hold time settings on the local and peer routers are different, the smaller setting is used. If the hold time is 0, BGP does not send KEEPALIVE messages to its peers and never tears down the session.
- If the keepalive interval is not 0, the actual keepalive interval is the smaller one between 1/3 of the hold time and the keepalive interval.

Restrictions and guidelines

The hold time must be a minimum of three times the keepalive interval.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

ip vpn-instance *vpn-instance-name*

3. Configure the keepalive interval and hold time.
   - Configure the global keepalive interval and hold time.

     timer keepalive *keepalive* **hold** *holdtime*

     This command takes effect for new BGP sessions and does not affect existing sessions.
   - Configure the keepalive interval and hold time for a peer or peer group.

     peer { *group-name* | *ipv4-address* [*mask-length*] } **timer keepalive** *keepalive* **hold** *holdtime*

   By default, the keepalive interval is 60 seconds, and hold time is 180 seconds.

   The timers configured with the **timer** and **peer timer** commands do not take effect until a session is re-established (for example, a session is reset).

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Configure the keepalive interval and hold time.
   - Configure the global keepalive interval and hold time.

     timer keepalive *keepalive* **hold** *holdtime*

     This command takes effect for new BGP sessions and does not affect existing sessions.
   - Configure the keepalive interval and hold time for a peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **timer keepalive** *keepalive* **hold** *holdtime*

   By default, the keepalive interval is 60 seconds, and hold time is 180 seconds.

   The timers configured with the **timer** and **peer timer** commands do not take effect until a session is re-established (for example, a session is reset).

Setting the session retry timer

About this task

To speed up session establishment to a peer or peer group and route convergence, set a small session retry timer. If the BGP session flaps, you can set a large session retry timer to reduce the impact.

Restrictions and guidelines

The timer set by the **peer timer connect-retry** command takes precedence over the timer set by the **timer connect-retry** command.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.

- Enter BGP instance view.

  bgp *as-number* [**instance** *instance-name*]
- Execute the following commands in sequence to enter BGP-VPN instance view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

3. Set the session retry timer.
   - Set the session retry timer for all peers or peer groups.

     timer connect-retry *retry-time*
   - Set the session retry timer for a peer or peer group.

     peer { *group-name* | *ipv4-address* [*mask-length*] } **timer connect-retry** *retry-time*

   By default, the session retry timer is 32 seconds for a peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Set the session retry timer.
   - Set the session retry timer for all peers or peer groups.

     timer connect-retry *retry-time*
   - Set the session retry timer for a peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **timer connect-retry** *retry-time*

   By default, the session retry timer is 32 seconds for a peer or peer group.

Configuring the interval for sending updates for the same route

About this task

A BGP router sends an UPDATE message to its peers when a route is changed. If the route changes frequently, the BGP router keeps sending updates for the same route, resulting route flapping. To prevent this situation, perform this task to configure the interval for sending updates for the same route to a peer or peer group.

This feature does not take effect on withdrawn routes. For withdrawn routes, BGP sends the withdrawal messages immediately.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

bgp *as-number* [**instance** *instance-name*]

- Execute the following commands in sequence to enter BGP-VPN instance view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

3. Configure the interval for sending updates for the same route to a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **route-update-interval** *interval*

   By default, the interval is 15 seconds for an IBGP peer and 30 seconds for an EBGP peer.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure the interval for sending updates for the same route to a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **route-update-interval** *interval*

   By default, the interval is 15 seconds for an IBGP peer and 30 seconds for an EBGP peer.

Configuring BGP logging and notifications

Enabling logging for session state changes

About this task

Perform this task to enable BGP to log BGP session establishment and disconnection events. To display the log information, use the **display bgp peer ipv4 unicast log-info** command or the **display bgp peer ipv6 unicast log-info** command. The logs are sent to the information center. The output rules of the logs (whether to output the logs and where to output) are determined by the information center configuration.

For more information about information center configuration, see *Network Management and Monitoring Configuration Guide*.

Procedure (IPv4 unicast/IPv4 multicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable logging for session state changes globally.

   log-peer-change

   By default, logging for session state changes is enabled globally.

4. (Optional.) Enter BGP-VPN instance view.

   ip vpn-instance *vpn-instance-name*

5. Enable logging for session state changes for a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **log-change**

   By default, logging for session state changes is enabled for all peers and peer groups.

Procedure (IPv6 unicast/IPv6 multicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable logging for session state changes globally.

   log-peer-change

   By default, logging for session state changes is enabled globally.

4. (Optional.) Enter BGP-VPN instance view.

   ip vpn-instance *vpn-instance-name*

5. Enable logging for session state changes for a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **log-change**

   By default, logging for session state changes is enabled for all peers and peer groups.

Configuring BGP network management

About this task

After you enable SNMP notifications for BGP, the device generates a notification when a BGP neighbor state change occurs. The notification includes the neighbor address, the error code and subcode of the most recent error, and the current neighbor state. For BGP notifications to be sent correctly, you must also configure SNMP on the device.

BGP does not know the BGP instance to which a managed MIB node belongs. To resolve this issue, configure different SNMP contexts for different BGP instances.

The device selects a MIB for an SNMP packet according to the context (for SNMPv3) or community name (for SNMPv1/v2c) in the following ways:

- For an SNMPv3 packet:
  - The device selects the MIB of the default BGP instance if the packet does not carry a context and no SNMP context is configured for the default BGP instance.
  - The device selects the MIB of a BGP instance if the packet meets the following conditions:
    - Carries a context that is configured with the **snmp-agent context** command in system view.
    - Matches the context of the BGP instance.
  - The device does not process any MIBs in other situations.
- For an SNMPv1/v2c packet:
  - The device selects the MIB of the default BGP instance if the following conditions are met:
    - No community name-to-SNMP context mapping is configured with the **snmp-agent community-map** command in system view.
    - No SNMP context is configured for the default BGP instance.
  - The device selects the MIB of a BGP instance if the community name is mapped to an SNMP context and the context matches the context of the BGP instance.
  - The device does not process any MIBs in other situations.

For more information about SNMP contexts and community names, see SNMP configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.
   system-view
2. Enable SNMP notifications for BGP.
   snmp-agent trap enable bgp [**instance** *instance-name*]
   By default, SNMP notifications for BGP are enabled.
3. Enter BGP instance view.
   bgp *as-number* [**instance** *instance-name*]
4. Configure an SNMP context for the BGP instance.
   snmp context-name *context-name*
   By default, no SNMP context is configured for the BGP instance.

Display and maintenance commands for basic BGP network building

Displaying BGP

Execute **display** commands in any view.

Displaying BGP (IPv4 unicast address family)

Task	Command
Display BGP IPv4 unicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [**group-name** *group-name*]
Display information about routes advertised by the **network** command and shortcut routes configured by the **network short-cut** command.	**display bgp** [**instance** *instance-name*] **network ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv4 unicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* *mask-length* | { *ipv4-address* | **group-name** *group-name* } **log-info** | [*ipv4-address*] **verbose**]
Display BGP IPv4 unicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* [{ *mask-length* | *mask* } [**longest-match**]] | **as-path-acl** *as-path-acl-number*]
Display BGP IPv4 unicast routing information.	**display bgp** [**instance** *instance-name*] **routing-table ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* [{ *mask-length* | *mask* }

Task	Command
	[**longest-match**]] | *ipv4-address* [*mask-length* | *mask*] **advertise-info** | **as-path-acl** *as-path-acl-number* | **community-list** { { *basic-community-list-number* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **peer** *ipv4-address* { **advertised-routes** | **received-routes** } [*ipv4-address* [*mask-length* | *mask*] | **statistics**] | **statistics**]
Display BGP IPv4 unicast address family update group information.	**display bgp** [**instance** *instance-name*] **update-group ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address*]
Display information about all BGP instances.	**display bgp instance-info**

Displaying BGP (IPv6 unicast address family)

Task	Command
Display BGP IPv6 unicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [**group-name** *group-name*]
Display information about routes advertised by the **network** command and shortcut routes configured by the **network short-cut** command.	**display bgp** [**instance** *instance-name*] **network ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv6 unicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* | { *ipv6-address* | **group-name** *group-name* } **log-info** | [*ipv6-address*] **verbose**] **display bgp** [**instance** *instance-name*] **peer ipv6** [**unicast**] [*ipv4-address* *mask-length* | *ipv4-address* **log-info** | [*ipv4-address*] **verbose**]
Display BGP IPv6 unicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* | **as-path-acl** *as-path-acl-number*]
Display BGP IPv6 unicast routing information.	**display bgp** [**instance** *instance-name*] **routing-table ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* [**advertise-info**] | **as-path-acl**

Task	Command
	as-path-acl-number | **community-list** { { *basic-community-list-number* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **peer** *ipv6-address* { **advertised-routes** | **received-routes** } [*ipv6-address prefix-length* | **statistics**] | **statistics**] **display bgp** [**instance** *instance-name*] **routing-table ipv6** [**unicast**] **peer** *ipv4-address* { **advertised-routes** | **received-routes** } [*ipv6-address prefix-length* | **statistics**]
Display BGP IPv6 unicast address family update group information.	**display bgp** [**instance** *instance-name*] **update-group ipv6** [**unicast**] [*ipv4-address* | *ipv6-address*] **display bgp** [**instance** *instance-name*] **update-group ipv6** [**unicast**] **vpn-instance** *vpn-instance-name* [*ipv6-address*]
Display information about all BGP instances.	**display bgp instance-info**

Displaying BGP (IPv4 multicast address family)

Task	Command
Display BGP IPv4 multicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv4 multicast** [**group-name** *group-name*]
Display information about routes advertised by the **network** command and shortcut routes configured by the **network short-cut** command.	**display bgp** [**instance** *instance-name*] **network ipv4 multicast**
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]
Display BGP IPv4 multicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv4 multicast** [*ipv4-address mask-length* | { *ipv4-address* | **group-name** *group-name* } **log-info** | [*ipv4-address*] **verbose**]
Display BGP IPv4 multicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv4 multicast** [*ipv4-address* [{ *mask-length* | *mask* } [**longest-match**]] | **as-path-acl** *as-path-acl-number*]
Display BGP IPv4 multicast routing information.	**display bgp** [**instance** *instance-name*] **routing-table ipv4 multicast** [*ipv4-address* [{ *mask-length* | *mask* } [**longest-match**]] | *ipv4-address* [*mask-length* | *mask*] **advertise-info** |

Task	Command
	as-path-acl *as-path-acl-number* | **community-list** { { *basic-community-list-number* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **peer** *ipv4-address* { **advertised-routes** | **received-routes** } [*ipv4-address* [*mask-length* | *mask*] | **statistics**] | **statistics**]
Display BGP IPv4 multicast address family update group information.	**display bgp** [**instance** *instance-name*] **update-group ipv4 multicast** [*ipv4-address*]
Display information about all BGP instances.	**display bgp instance-info**

Displaying BGP (IPv6 multicast address family)

Task	Command
Display BGP IPv6 multicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv6 multicast** [**group-name** *group-name*]
Display information about routes advertised by the **network** command and shortcut routes configured by the **network short-cut** command.	**display bgp** [**instance** *instance-name*] **network ipv6 multicast**
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]
Display BGP IPv6 multicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv6 multicast** [*ipv6-address prefix-length* | { *ipv6-address* | **group-name** *group-name* } **log-info** | [*ipv6-address*] **verbose**]
Display BGP IPv6 multicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv6 multicast** [*ipv6-address prefix-length* | **as-path-acl** *as-path-acl-number*]
Display BGP IPv6 multicast routing information.	**display bgp** [**instance** *instance-name*] **routing-table ipv6 multicast** [*ipv6-address prefix-length* [**advertise-info**] | **as-path-acl** *as-path-acl-number* | **community-list** { { *basic-community-list-number* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **peer** *ipv6-address* { **advertised-routes** | **received-routes** } [*ipv6-address prefix-length* | **statistics**] | **statistics**]
Display BGP IPv6 multicast address family update group information.	**display bgp** [**instance** *instance-name*] **update-group ipv6 multicast**

Task	Command
	[*ipv6-address*]
Display information about all BGP instances.	**display bgp instance-info**

Resetting BGP sessions

CAUTION:

A reset operation tears down BGP sessions for a short period of time.

Execute **reset** commands in user view.

Task	Command
Reset BGP sessions for IPv4 unicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv4-address* [*mask-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Reset BGP sessions for IPv4 multicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv4-address* [*mask-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv4 multicast**
Reset BGP sessions for IPv6 unicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] **reset bgp** *ipv4-address* [*mask-length*] **ipv6** [**unicast**]
Reset BGP sessions for IPv6 multicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv6 multicast**
Reset all BGP sessions.	**reset bgp** [**instance** *instance-name*] **all**

Clearing BGP information

Execute **reset** commands in user view.

Task	Command
Clear flap information for BGP IPv4 unicast routes.	**reset bgp** [**instance** *instance-name*] **flap-info ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* [*mask-length* | *mask*] | **as-path-acl** *as-path-acl-number* | **peer** *ipv4-address* [*mask-length*]]
Clear flap information for BGP IPv4	**reset bgp** [**instance** *instance-name*]

Task	Command
multicast routes.	**flap-info ipv4 multicast** [*ipv4-address* [*mask-length* | *mask*] | **as-path-acl** *as-path-acl-number* | **peer** *ipv4-address* [*mask-length*]]
Clear flap information for BGP IPv6 unicast routes.	**reset bgp** [**instance** *instance-name*] **flap-info ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* | **as-path-acl** *as-path-acl-number* | **peer** *ipv6-address* [*prefix-length*]]
Clear flap information for BGP IPv6 multicast routes.	**reset bgp** [**instance** *instance-name*] **flap-info ipv6 multicast** [*ipv6-address* *prefix-length* | **as-path-acl** *as-path-acl-number* | **peer** *ipv6-address* [*prefix-length*]]

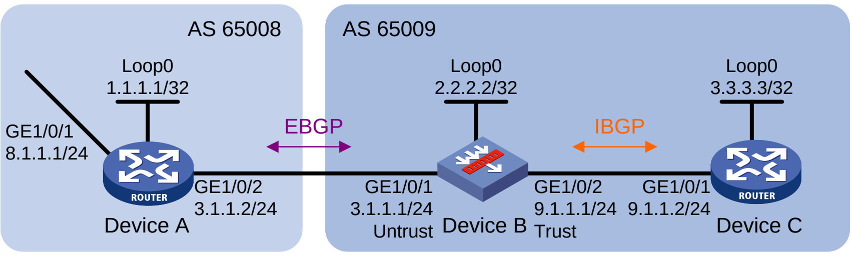
Basic IPv4 BGP network configuration examples

Example: Configuring basic BGP

Network configuration

As shown in Figure 10, all devices run BGP. Run EBGP between Device A and Device B, and run IBGP between Device B and Device C to allow Device C to access network 8.1.1.0/24 connected to Device A.

Figure 10 Network diagram



Procedure

1. Configure IP addresses for interfaces. (Details not shown.)
2. Add each interface on Device B to a security zone according to Figure 10.

```
<DeviceB> system-view
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.
   a. Configure a security policy to allow EBGP session establishment between Device A and Device B by permitting traffic between security zone **untrust** and security zone **local**.

# Create security policy rule **ebgplocalin** and permit Device B to receive BGP packets from Device A.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name ebgplocalin
[DeviceB-security-policy-ip-0-ebgplocalin] source-zone untrust
[DeviceB-security-policy-ip-0-ebgplocalin] destination-zone local
[DeviceB-security-policy-ip-0-ebgplocalin] service bgp
[DeviceB-security-policy-ip-0-ebgplocalin] action pass
[DeviceB-security-policy-ip-0-ebgplocalin] quit
```

# Create security policy rule **ebgplocalout** and permit Device B to send BGP packets to Device A.

```
[DeviceB-security-policy-ip] rule name ebgplocalout
[DeviceB-security-policy-ip-1-ebgplocalout] source-zone local
[DeviceB-security-policy-ip-1-ebgplocalout] destination-zone untrust
[DeviceB-security-policy-ip-1-ebgplocalout] service bgp
[DeviceB-security-policy-ip-1-ebgplocalout] action pass
[DeviceB-security-policy-ip-1-ebgplocalout] quit
```

b. Configure a security policy to allow IBGP session establishment between Device B and Device C by permitting traffic between security zone **trust** and security zone **local**.

# Create security policy rule **bgplocalout** and permit Device B to send BGP and OSPF packets to Device C.

```
[DeviceB-security-policy-ip] rule name bgplocalout
[DeviceB-security-policy-ip-2-bgplocalout] source-zone local
[DeviceB-security-policy-ip-2-bgplocalout] destination-zone trust
[DeviceB-security-policy-ip-2-bgplocalout] service bgp
[DeviceB-security-policy-ip-2-bgplocalout] service ospf
[DeviceB-security-policy-ip-2-bgplocalout] action pass
[DeviceB-security-policy-ip-2-bgplocalout] quit
```

# Create security policy rule **bgplocalin** and permit Device B to receive BGP and OSPF packets from Device C.

```
[DeviceB-security-policy-ip] rule name bgplocalin
[DeviceB-security-policy-ip-3-bgplocalin] source-zone trust
[DeviceB-security-policy-ip-3-bgplocalin] destination-zone local
[DeviceB-security-policy-ip-3-bgplocalin] service bgp
[DeviceB-security-policy-ip-3-bgplocalin] service ospf
[DeviceB-security-policy-ip-3-bgplocalin] action pass
[DeviceB-security-policy-ip-3-bgplocalin] quit
```

c. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust** so that Device C can access network 8.1.1.0/24 connected to Device A.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-4-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-4-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-4-trust-untrust] source-ip-subnet 9.1.1.0 24
[DeviceB-security-policy-ip-4-trust-untrust] destination-ip-subnet 8.1.1.0 24
[DeviceB-security-policy-ip-4-trust-untrust] action pass
[DeviceB-security-policy-ip-4-trust-untrust] quit
[DeviceB-security-policy-ip] quit
```

4. Configure IBGP:

   - To prevent route flapping caused by port state changes, this example uses loopback interfaces to establish IBGP connections.
   - Because loopback interfaces are virtual interfaces, you need to use the **peer connect-interface** command to specify the loopback interface as the source interface for BGP connection establishment.
   - Enable OSPF in AS 65009 to ensure that Device B can communicate with Device C through loopback interfaces.

# Configure Device B.

```
[DeviceB] bgp 65009
[DeviceB-bgp-default] router-id 2.2.2.2
[DeviceB-bgp-default] peer 3.3.3.3 as-number 65009
[DeviceB-bgp-default] peer 3.3.3.3 connect-interface loopback 0
[DeviceB-bgp-default] address-family ipv4 unicast
[DeviceB-bgp-default-ipv4] peer 3.3.3.3 enable
[DeviceB-bgp-default-ipv4] quit
[DeviceB-bgp-default] quit
[DeviceB] ospf 1
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
[DeviceB-ospf-1-area-0.0.0.0] network 9.1.1.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] quit
```

# Configure Device C.

```
<DeviceC> system-view
[DeviceC] bgp 65009
[DeviceC-bgp-default] router-id 3.3.3.3
[DeviceC-bgp-default] peer 2.2.2.2 as-number 65009
[DeviceC-bgp-default] peer 2.2.2.2 connect-interface loopback 0
[DeviceC-bgp-default] address-family ipv4 unicast
[DeviceC-bgp-default-ipv4] peer 2.2.2.2 enable
[DeviceC-bgp-default-ipv4] quit
[DeviceC-bgp-default] quit
[DeviceC] ospf 1
[DeviceC-ospf-1] area 0
[DeviceC-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[DeviceC-ospf-1-area-0.0.0.0] network 9.1.1.0 0.0.0.255
[DeviceC-ospf-1-area-0.0.0.0] quit
[DeviceC-ospf-1] quit
[DeviceC] display bgp peer ipv4

 BGP local router ID : 3.3.3.3
 Local AS number : 65009
 Total number of peers : 1                 Peers in established state : 1

  * - Dynamically created peer
  Peer                    AS  MsgRcvd  MsgSent OutQ PrefRcv Up/Down  State

  2.2.2.2              65009        2        2    0       0 00:00:13 Established
```

The output shows that Device C has established an IBGP peer relationship with Device B.

5. Configure EBGP:
    - The EBGP peers (usually in different ISPs) are located in different ASs. Typically, their loopback interfaces are not reachable to each other, so directly connected interfaces are used for establishing EBGP sessions.
    - To enable Device C to access the network 8.1.1.0/24 connected directly to Device A, inject network 8.1.1.0/24 to the BGP routing table of Device A.

# Configure Device A.

```
<DeviceA> system-view
[DeviceA] bgp 65008
[DeviceA-bgp-default] router-id 1.1.1.1
[DeviceA-bgp-default] peer 3.1.1.1 as-number 65009
[DeviceA-bgp-default] address-family ipv4 unicast
[DeviceA-bgp-default-ipv4] peer 3.1.1.1 enable
[DeviceA-bgp-default-ipv4] network 8.1.1.0 24
[DeviceA-bgp-default-ipv4] quit
[DeviceA-bgp-default] quit
```

# Configure Device B.

```
[DeviceB] bgp 65009
[DeviceB-bgp-default] peer 3.1.1.2 as-number 65008
[DeviceB-bgp-default] address-family ipv4 unicast
[DeviceB-bgp-default-ipv4] peer 3.1.1.2 enable
[DeviceB-bgp-default-ipv4] quit
[DeviceB-bgp-default] quit
```

# Display BGP peer information on Device B.

```
[DeviceB] display bgp peer ipv4

 BGP local router ID : 2.2.2.2
 Local AS number : 65009
 Total number of peers : 2                 Peers in established state : 2

  * - Dynamically created peer
  Peer                    AS  MsgRcvd  MsgSent OutQ PrefRcv Up/Down  State

  3.3.3.3              65009        4        4    0       0 00:02:49 Established
  3.1.1.2              65008        2        2    0       0 00:00:05 Established
```

The output shows that Device B has established an IBGP peer relationship with Device C and an EBGP peer relationship with Device A.

# Display the BGP routing table on Device A.

```
[DeviceA] display bgp routing-table ipv4

 Total number of routes: 1

 BGP local router ID is 1.1.1.1
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete
```

```
     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

* >  8.1.1.0/24         8.1.1.1         0                     32768   i
```

# Display the BGP routing table on Device B.

```
[DeviceB] display bgp routing-table ipv4

 Total number of routes: 1

 BGP local router ID is 2.2.2.2
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete

     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

* >e 8.1.1.0/24         3.1.1.2         0                     0       65008i
```

# Display the BGP routing table on Device C.

```
[DeviceC] display bgp routing-table ipv4

 Total number of routes: 1

 BGP local router ID is 3.3.3.3
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete

     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

   i 8.1.1.0/24         3.1.1.2         0          100        0       65008i
```

The outputs show that Device A has no route to AS 65009, and Device C has learned network 8.1.1.0, but the next hop 3.1.1.2 is unreachable. As a result, the route is invalid.

6. Redistribute direct routes:

# Configure BGP to redistribute direct routes on Device B, so Device A can obtain the route to 9.1.1.0/24, and Device C can obtain the route to 3.1.1.0/24.

```
[DeviceB] bgp 65009
[DeviceB-bgp-default] address-family ipv4 unicast
[DeviceB-bgp-default-ipv4] import-route direct
[DeviceB-bgp-default-ipv4] quit
[DeviceB-bgp-default] quit
```

# Display the BGP routing table on Device A.

```
[DeviceA] display bgp routing-table ipv4

 Total number of routes: 4

 BGP local router ID is 1.1.1.1
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete
```

```
     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

* >e 2.2.2.2/32         3.1.1.1         0                     0       65009?
* >e 3.1.1.0/24         3.1.1.1         0                     0       65009?
* >  8.1.1.0/24         8.1.1.1         0                     32768   i
* >e 9.1.1.0/24         3.1.1.1         0                     0       65009?
```

Two routes 2.2.2.2/32 and 9.1.1.0/24 have been added in Device A's routing table.

# Display the BGP routing table on Device C.

```
[DeviceC] display bgp routing-table ipv4

 Total number of routes: 4

 BGP local router ID is 3.3.3.3
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete

     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

* >i 2.2.2.2/32         2.2.2.2         0          100        0       ?
* >i 3.1.1.0/24         2.2.2.2         0          100        0       ?
* >i 8.1.1.0/24         3.1.1.2         0          100        0       65008i
* >i 9.1.1.0/24         2.2.2.2         0          100        0       ?
```

The output shows that the route 8.1.1.0 has become valid and the next hop is Device A.

Verifying the configuration

# Verify that Device C can ping 8.1.1.1.

```
[DeviceC] ping 8.1.1.1
Ping 8.1.1.1 (8.1.1.1): 56 data bytes, press CTRL+C to break
56 bytes from 8.1.1.1: icmp_seq=0 ttl=255 time=2.000 ms
56 bytes from 8.1.1.1: icmp_seq=1 ttl=255 time=0.000 ms
56 bytes from 8.1.1.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 8.1.1.1: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 8.1.1.1: icmp_seq=4 ttl=255 time=1.000 ms

--- Ping statistics for 8.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/2.000/0.800 ms
```

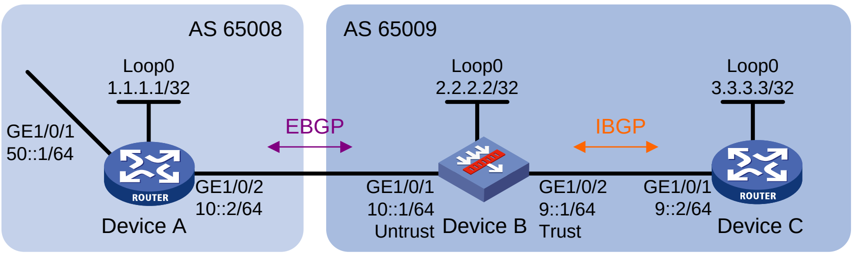
Basic IPv6 BGP network configuration examples

Example: Configuring IPv6 basic BGP

Network configuration

As shown in Figure 11, all devices run BGP. Run EBGP between Device A and Device B, and run IBGP between Device B and Device C to allow Device C to access network 50::/64 connected to Device A.

Figure 11 Network diagram



Procedure

1. Configure IPv6 addresses for interfaces and IPv4 addresses for Loopback interfaces. (Details not shown.)
2. Add each interface on Device B to a security zone according to Figure 11.

```
<DeviceB> system-view
[DeviceB] security-zone name untrust
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] quit
```

3. Configure security policies.

   a. Configure a security policy to allow EBGP session establishment between Device A and Device B by permitting traffic between security zone **untrust** and security zone **local**.

   # Create security policy rule **ebgplocalin** and permit Device B to receive BGP packets from Device A.

```
[DeviceB] security-policy ipv6
[DeviceB-security-policy-ipv6] rule name ebgplocalin
[DeviceB-security-policy-ipv6-0-ebgplocalin] source-zone untrust
[DeviceB-security-policy-ipv6-0-ebgplocalin] destination-zone local
[DeviceB-security-policy-ipv6-0-ebgplocalin] service bgp
[DeviceB-security-policy-ipv6-0-ebgplocalin] action pass
[DeviceB-security-policy-ipv6-0-ebgplocalin] quit
```

   # Create security policy rule **ebgplocalout** and permit Device B to send BGP packets to Device A.

```
[DeviceB-security-policy-ipv6] rule name ebgplocalout
[DeviceB-security-policy-ipv6-1-ebgplocalout] source-zone local
[DeviceB-security-policy-ipv6-1-ebgplocalout] destination-zone untrust
[DeviceB-security-policy-ipv6-1-ebgplocalout] service bgp
[DeviceB-security-policy-ipv6-1-ebgplocalout] action pass
[DeviceB-security-policy-ipv6-1-ebgplocalout] quit
```

   b. Configure a security policy to allow IBGP session establishment between Device B and Device C by permitting traffic between security zone **trust** and security zone **local**.

   # Create security policy rule **bgplocalout** and permit Device B to send BGP packets to Device C.

```
[DeviceB-security-policy-ipv6] rule name bgplocalout
[DeviceB-security-policy-ipv6-2-bgplocalout] source-zone local
[DeviceB-security-policy-ipv6-2-bgplocalout] destination-zone trust
[DeviceB-security-policy-ipv6-2-bgplocalout] service bgp
```

```
[DeviceB-security-policy-ipv6-2-bgplocalout] action pass
[DeviceB-security-policy-ipv6-2-bgplocalout] quit
```

c. Configure a security policy to permit traffic between security zone **untrust** and security zone **trust** so that Device C can access network 50::/64 connected to Device A.

# Create security policy rule **trust-untrust** and permit packets from security zone **trust** to security zone **untrust** to pass.

```
[DeviceB-security-policy-ipv6] rule name trust-untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-zone trust
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ipv6-3-trust-untrust] source-ip-subnet 9:: 64
[DeviceB-security-policy-ipv6-3-trust-untrust] destination-ip-subnet 50:: 64
[DeviceB-security-policy-ipv6-3-trust-untrust] action pass
[DeviceB-security-policy-ipv6-3-trust-untrust] quit
[DeviceB-security-policy-ipv6] quit
```

4. Configure IBGP:

# Configure Device B.

```
［DeviceB］ bgp 65009
［DeviceB-bgp-default］ router-id 2.2.2.2
［DeviceB-bgp-default］ peer 9::2 as-number 65009
［DeviceB-bgp-default］ address-family ipv6
［DeviceB-bgp-default-ipv6］ peer 9::2 enable
［DeviceB-bgp-default-ipv6］ quit
```

# Configure Device C.

```
<DeviceC> system-view
［DeviceC］ bgp 65009
［DeviceC-bgp-default］ router-id 3.3.3.3
［DeviceC-bgp-default］ peer 9::1 as-number 65009
［DeviceC-bgp-default］ address-family ipv6
［DeviceC-bgp-default-ipv6］ peer 9::1 enable
```

5. Configure EBGP:

# Configure Device A.

```
<DeviceA> system-view
［DeviceA］ bgp 65008
［DeviceA-bgp-default］ router-id 1.1.1.1
［DeviceA-bgp-default］ peer 10::1 as-number 65009
［DeviceA-bgp-default］ address-family ipv6
［DeviceA-bgp-default-ipv6］ peer 10::1 enable
```

# Configure Device B.

```
［DeviceB-bgp-default］ peer 10::2 as-number 65008
［DeviceB-bgp-default］ address-family ipv6
［DeviceB-bgp-default-ipv6］ peer 10::2 enable
```

6. Inject network routes to the BGP routing table:

# Configure Device A.

```
［DeviceA-bgp-default-ipv6］ network 10:: 64
［DeviceA-bgp-default-ipv6］ network 50:: 64
［DeviceA-bgp-default-ipv6］ quit
［DeviceA-bgp-default］ quit
```

# Configure Device B.

```
[DeviceB-bgp-default-ipv6] network 10:: 64
[DeviceB-bgp-default-ipv6] network 9:: 64
[DeviceB-bgp-default-ipv6] quit
[DeviceB-bgp-default] quit
```

# Configure Device C.

```
[DeviceC-bgp-default-ipv6] network 9:: 64
[DeviceC-bgp-default-ipv6] quit
[DeviceC-bgp-default] quit
```

Verifying the configuration

# Display IPv6 BGP peer information on Device B.

```
[DeviceB] display bgp peer ipv6

 BGP local router ID: 2.2.2.2
 Local AS number: 65009
 Total number of peers: 2                  Peers in established state: 2

  * - Dynamically created peer
  Peer                    AS  MsgRcvd  MsgSent OutQ PrefRcv Up/Down  State

  9::2                 65009       41       43    0       1 00:29:00 Established
  10::2                65008       38       38    0       2 00:27:20 Established
```

The output shows that Device A and Device B have established an EBGP connection, and Device B and Device C have established an IBGP connection.

# Display IPv6 BGP routing table information on Device A.

```
[DeviceA] display bgp routing-table ipv6

 Total number of routes: 4

 BGP local router ID is 1.1.1.1
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete

* >e Network : 9::                                      PrefixLen : 64
     NextHop : 10::1                                    LocPrf    :
     PrefVal : 0                                        OutLabel  : NULL
     MED     : 0
     Path/Ogn: 65009i

* >  Network : 10::                                     PrefixLen : 64
     NextHop : ::                                       LocPrf    :
     PrefVal : 32768                                    OutLabel  : NULL
     MED     : 0
     Path/Ogn: i

*  e Network : 10::                                     PrefixLen : 64
     NextHop : 10::1                                    LocPrf    :
```

```
     PrefVal : 0                                          OutLabel  : NULL
     MED     : 0
     Path/Ogn: 65009i

* >  Network : 50::                                       PrefixLen : 64
     NextHop : ::                                         LocPrf    :
     PrefVal : 32768                                      OutLabel  : NULL
     MED     : 0
     Path/Ogn: i
```

The output shows that Device A has learned routing information of AS 65009.

# Display IPv6 BGP routing table information on Device C.

```
[DeviceC] display bgp routing-table ipv6

 Total number of routes: 4

 BGP local router ID is 3.3.3.3
 Status codes: * - valid, > - best, d - dampened, h - history,
               s - suppressed, S - stale, i - internal, e - external
               Origin: i - IGP, e - EGP, ? - incomplete

* >  Network : 9::                                        PrefixLen : 64
     NextHop : ::                                         LocPrf    :
     PrefVal : 32768                                      OutLabel  : NULL
     MED     : 0
     Path/Ogn: i

*   i Network : 9::                                       PrefixLen : 64
     NextHop : 9::1                                       LocPrf    : 100
     PrefVal : 0                                          OutLabel  : NULL
     MED     : 0
     Path/Ogn: i

* >i Network : 10::                                       PrefixLen : 64
     NextHop : 9::1                                       LocPrf    : 100
     PrefVal : 0                                          OutLabel  : NULL
     MED     : 0
     Path/Ogn: i

* >i Network : 50::                                       PrefixLen : 64
     NextHop : 10::2                                      LocPrf    : 100
     PrefVal : 0                                          OutLabel  : NULL
     MED     : 0
     Path/Ogn: 65008i
```

The output shows that Device C has learned the route 50::/64.

# Verify that Device C can ping hosts on network 50::/64.

```
[DeviceC]ping ipv6 50::1
Ping6(56 data bytes) 9::2 --> 50::1, press CTRL+C to break
```

```
56 bytes from 50::1, icmp_seq=0 hlim=63 time=1.000 ms
56 bytes from 50::1, icmp_seq=1 hlim=63 time=1.000 ms
56 bytes from 50::1, icmp_seq=2 hlim=63 time=2.000 ms
56 bytes from 50::1, icmp_seq=3 hlim=63 time=1.000 ms
56 bytes from 50::1, icmp_seq=4 hlim=63 time=1.000 ms

--- Ping6 statistics for 50::1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/1.200/2.000/0.400 ms
```

Troubleshooting BGP

State of the connection to a peer cannot become established

Symptom

The **display bgp peer ipv4 unicast** or **display bgp peer ipv6 unicast** command output shows that the state of the connection to a peer cannot become established.

Analysis

To become BGP peers, any two routers must establish a TCP connection using port 179 and exchange OPEN messages successfully.

Solution

1. To resolve the problem:
   a. Use the **display current-configuration** command to verify the current configuration, and verify that the peer's AS number is correct.
   b. Use the **display bgp peer ipv4 unicast** or **display bgp peer ipv6 unicast** command to verify that the peer's IP/IPv6 address is correct.
   c. If a loopback interface is used, verify that the loopback interface is specified with the **peer connect-interface** command.
   d. If the peer is a non-direct EBGP peer, verify that the **peer ebgp-max-hop** command is configured.
   e. If the **peer ttl-security hops** command is configured, verify that the command is configured on the peer. Verify that the *hop-count* values configured on them are greater than the number of hops between them.
   f. Verify that a valid route to the peer is available.
   g. Use the **ping** command to verify the connectivity to the peer.
   h. Use the **display tcp verbose** or **display ipv6 tcp verbose** command to verify the TCP connection.
   i. Verify that no ACL rule is applied to disable TCP port 179.
2. If the problem persists, contact NSFOCUS Support.

Configuring large-scale BGP networks

Large-scale BGP network configuration tasks at a glance

To configure large-scale BGP networks, perform the following tasks:

- Configuring BGP route dampening
- Configuring BGP communities
- Configuring BGP route reflection
- Configuring BGP confederation settings
  - Configuring a BGP confederation
  - (Optional.) Configuring confederation compatibility

Configuring BGP route dampening

About this task

Route dampening enables BGP to not select unstable routes as optimal routes.

Restrictions and guidelines

This feature applies to EBGP routes but not to IBGP routes.

If an EBGP peer goes down after you configure this feature, routes coming from the peer are dampened but not deleted.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure BGP route dampening.

   dampening [*half-life-reachable half-life-unreachable reuse suppress ceiling* | **route-policy** *route-policy-name*] *

By default, BGP route dampening is not configured.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Configure IPv6 BGP route dampening.

   dampening [*half-life-reachable half-life-unreachable reuse suppress ceiling* | **route-policy** *route-policy-name*] *

   By default, IPv6 BGP route dampening is not configured.

Configuring BGP communities

About this task

By default, a router does not advertise the COMMUNITY or extended community attribute to its peers or peer groups. When the router receives a route carrying the COMMUNITY or extended community attribute, it removes the attribute before advertising the route to other peers or peer groups.

Perform this task to enable a router to advertise the COMMUNITY or extended community attribute to its peers for route filtering and control. You can also use a routing policy to add or modify the COMMUNITY or extended community attribute for specific routes. For more information about routing policy, see "Configuring routing policies."

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

```
bgp as-number [ instance instance-name ]
ip vpn-instance vpn-instance-name
address-family ipv4 [ unicast ]
```

- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  ```
  bgp as-number [ instance instance-name ]
  address-family ipv4 multicast
  ```

3. Advertise the COMMUNITY attribute to a peer or peer group.

   ```
   peer { group-name | ipv4-address [ mask-length ] } advertise-community
   ```

   By default, the COMMUNITY attribute is not advertised.

4. Advertise the extended community attribute to a peer or peer group.

   ```
   peer { group-name | ipv4-address [ mask-length ] }
   advertise-ext-community
   ```

   By default, the extended community attribute is not advertised.

5. (Optional.) Apply a routing policy to routes advertised to a peer or peer group.

   ```
   peer { group-name | ipv4-address [ mask-length ] } route-policy
   route-policy-name export
   ```

   By default, no routing policy is applied.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv6 [ unicast ]
     ```

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     address-family ipv6 [ unicast ]
     ```

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     ```
     bgp as-number [ instance instance-name ]
     address-family ipv6 multicast
     ```

3. Advertise the COMMUNITY attribute to a peer or peer group.

   ```
   peer { group-name | ipv6-address [ prefix-length ] }
   advertise-community
   ```

   By default, the COMMUNITY attribute is not advertised.

4. Advertise the extended community attribute to a peer or peer group.

   ```
   peer { group-name | ipv6-address [ prefix-length ] }
   advertise-ext-community
   ```

   By default, the extended community attribute is not advertised.

5. (Optional.) Apply a routing policy to routes advertised to a peer or peer group.

peer { *group-name* | *ipv6-address* [*prefix-length*] } **route-policy** *route-policy-name* **export**

By default, no routing policy is applied.

Configuring BGP route reflection

Configuring a BGP route reflector

About this task

Perform this task to configure a BGP route reflector and its clients. The route reflector and its clients automatically form a cluster identified by the router ID of the route reflector. The route reflector forwards route updates among its clients.

To improve availability, you can specify multiple route reflectors for a cluster. The route reflectors in the cluster must have the same cluster ID to avoid routing loops.

When a route reflector connects to multiple clusters, you can configure different cluster IDs for different peers or peer groups.

You only need to configure BGP route reflection on the device that acts as a route reflector. Other devices do not need to know the role of the local device in route reflection.

After you configure a device as a route reflector, it advertises routes as follows:

- Advertises routes received from a non-client IBGP peer to all clients.
- Advertises routes received from an IBGP peer that acts as a client to all peers.
- Advertises routes received from an EBGP peer to all peers.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure the router as a route reflector and specify a peer or peer group as its client.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **reflect-client**

   By default, no route reflector or client is configured.

4. (Optional.) Enable route reflection between clients.

   reflect between-clients

By default, route reflection between clients is enabled.

5. (Optional.) Configure the cluster ID of the route reflector.

   reflector cluster-id { *cluster-id* | *ipv4-address* }

   By default, a route reflector uses its own router ID as the cluster ID.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Configure the router as a route reflector and specify a peer or peer group as its client.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **reflect-client**

   By default, no route reflector or client is configured.

4. (Optional.) Enable route reflection between clients.

   reflect between-clients

   By default, route reflection between clients is enabled.

5. (Optional.) Configure the cluster ID of the route reflector.

   reflector cluster-id { *cluster-id* | *ipv4-address* }

   By default, a route reflector uses its own router ID as the cluster ID.

Ignoring the ORIGINATOR_ID attribute

About this task

By default, BGP drops incoming route updates whose ORIGINATOR_ID attribute is the same as the local router ID. Some special networks such as firewall networks require BGP to accept such route updates. To meet the requirement, you must configure BGP to ignore the ORIGINATOR_ID attribute.

Restrictions and guidelines

Make sure this command does not result in a routing loop.

After you execute this command, BGP also ignores the CLUSTER_LIST attribute.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

ip vpn-instance *vpn-instance-name*

3. Ignore the ORIGINATOR_ID attribute.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **ignore-originatorid**

   By default, BGP does not ignore the ORIGINATOR_ID attribute.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Ignore the ORIGINATOR_ID attribute.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ignore-originatorid**

   By default, BGP does not ignore the ORIGINATOR_ID attribute.

Configuring BGP confederation settings

About BGP confederation

BGP confederation provides another way to reduce IBGP connections in an AS.

A confederation contains sub-ASs. In each sub-AS, IBGP peers are fully meshed. Sub-ASs establish EBGP connections in between.

Configuring a BGP confederation

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Configure a confederation ID.

   confederation id *as-number*

   By default, no confederation ID is configured.

   From an outsider's perspective, the sub-ASs of the confederation is a single AS, which is identified by the confederation ID.

4. Specify confederation peer sub-ASs in the confederation.

   confederation peer-as *as-number-list*

   By default, no confederation peer sub-ASs are specified.

   A confederation can contain a maximum of 32 sub-ASs. The AS number of a sub-AS is effective only in the confederation.

   If the router needs to establish EBGP connections to other sub-ASs, you must specify the peering sub-ASs in the confederation.

Configuring confederation compatibility

About this task

If any routers in the confederation do not comply with RFC 3065, enable confederation compatibility to allow the router to work with those routers.

Procedure

1. Enter system view.
   system-view
2. Enter BGP instance view.
   bgp *as-number* [**instance** *instance-name*]
3. Enable confederation compatibility.
   confederation nonstandard
   By default, confederation compatibility is disabled.

Display and maintenance commands for large-scale BGP network configuration

Execute **display** commands in any view.

Displaying BGP (IPv4 unicast address family)

Task	Command
Display BGP IPv4 unicast route dampening parameter information.	**display bgp** [**instance** *instance-name*] **dampening parameter ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv4 unicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [**group-name** *group-name*]
Display BGP IPv4 unicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* *mask-length* | { *ipv4-address* | **group-name** *group-name* } **log-info** | [*ipv4-address*] **verbose**]
Display dampened BGP IPv4 unicast route information.	**display bgp** [**instance** *instance-name*] **routing-table dampened ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv4 unicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv4-address* [{ *mask-length* | *mask* } [**longest-match**]] | **as-path-acl** *as-path-acl-number*]

Displaying BGP (IPv6 unicast address family)

Task	Command
Display BGP IPv6 unicast route dampening parameter information.	**display bgp** [**instance** *instance-name*] **dampening parameter ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv6 unicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [**group-name** *group-name*]
Display BGP IPv6 unicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* | { *ipv6-address* | **group-name** *group-name* } **log-info** | [*ipv6-address*] **verbose**] **display bgp** [**instance** *instance-name*] **peer ipv6** [**unicast**] [*ipv4-address* *mask-length* | *ipv4-address* **log-info** | [*ipv4-address*] **verbose**]
Display dampened BGP IPv6 unicast route information.	**display bgp** [**instance** *instance-name*] **routing-table dampened ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Display BGP IPv6 unicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] [*ipv6-address* *prefix-length* | **as-path-acl** *as-path-acl-number*]

Displaying BGP (IPv4 multicast address family)

Task	Command
Display BGP IPv4 multicast route dampening parameter information.	**display bgp** [**instance** *instance-name*] **dampening parameter ipv4 multicast**
Display BGP IPv4 multicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv4 multicast** [**group-name** *group-name*]
Display BGP IPv4 multicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv4 multicast** [*ipv4-address* *mask-length* | { *ipv4-address* | **group-name** *group-name* } **log-info** | [*ipv4-address*] **verbose**]
Display dampened BGP IPv4 multicast route information.	**display bgp** [**instance** *instance-name*] **routing-table dampened ipv4 multicast**
Display BGP IPv4 multicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv4 multicast** [*ipv4-address* [{ *mask-length* | *mask* } [**longest-match**]] | **as-path-acl** *as-path-acl-number*]

Displaying BGP (IPv6 multicast address family)

Task	Command
Display BGP IPv6 multicast route dampening parameter information.	**display bgp** [**instance** *instance-name*] **dampening parameter ipv6 multicast**
Display BGP IPv6 multicast peer group information.	**display bgp** [**instance** *instance-name*] **group ipv6 multicast** [**group-name** *group-name*]
Display BGP IPv6 multicast peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer ipv6 multicast** [*ipv6-address prefix-length* | { *ipv6-address* | **group-name** *group-name* } **log-info** | [*ipv6-address*] **verbose**]
Display dampened BGP IPv6 multicast route information.	**display bgp** [**instance** *instance-name*] **routing-table dampened ipv6 multicast**
Display BGP IPv6 multicast route flapping statistics.	**display bgp** [**instance** *instance-name*] **routing-table flap-info ipv6 multicast** [*ipv6-address prefix-length* | **as-path-acl** *as-path-acl-number*]

Controlling BGP path selection

BGP path selection control tasks at a glance

By configuring BGP path attributes, you can control BGP path selection.

To control BGP path selection, perform the following tasks:

1. Configuring preferences for BGP routes
2. Configuring the NEXT_HOP attribute
3. Setting a preferred value for received routes
4. Configuring the default local preference
5. Configuring the AS_PATH attribute
    - Permitting local AS number to appear in routes from a peer or peer group
    - Ignoring the AS_PATH attribute during optimal route selection
    - Advertising a fake AS number to a peer or peer group
    - Configuring AS number substitution
    - Removing private AS numbers from sent updates
    - Ignoring the first AS number of EBGP route updates
6. Configuring the MED attribute
    - Configuring the default MED value
    - Enabling MED comparison for routes from different ASs
    - Enabling MED comparison for routes on a per-AS basis
    - Enabling MED comparison for routes from confederation peers
7. Ignoring IGP metrics during optimal route selection
8. Ignoring router IDs during optimal route selection

Configuring preferences for BGP routes

About this task

Routing protocols each have a default preference. If they find multiple routes destined for the same network, the route found by the routing protocol with the highest preference is selected as the optimal route.

You can use the **preference** command to modify preferences for EBGP, IBGP, and local BGP routes, or use a routing policy to set a preference for matching routes. For routes not matching the routing policy, the default preference applies.

If a device has an EBGP route and a local BGP route to reach the same destination, it does not select the EBGP route because the EBGP route has a lower preference than the local BGP route by default. You can use the **network short-cut** command to configure the EBGP route as a shortcut route that has the same preference as the local BGP route. The EBGP route will more likely become the optimal route.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

- Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 [**unicast**]

- Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv4 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 multicast

3. Configure preferences for EBGP, IBGP, and local BGP routes.

   preference { *external-preference internal-preference local-preference* | **route-policy** *route-policy-name* }

   By default, the preferences for EBGP, IBGP, and local BGP routes are 255, 255, and 130, respectively.

4. (Optional.) Configure an EBGP route as a shortcut route.

   network *ipv4-address* [*mask-length* | *mask*] **short-cut**

   By default, an EBGP route has a preference of 255.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

3. Configure preferences for EBGP, IBGP, and local BGP routes.

   preference { *external-preference internal-preference local-preference* | **route-policy** *route-policy-name* }

   By default, the preferences for EBGP, IBGP, and local BGP routes are 255, 255, and 130, respectively.

4. (Optional.) Configure an EBGP route as a shortcut route.

   network *ipv6-address prefix-length* **short-cut**

By default, an EBGP route has a preference of 255.

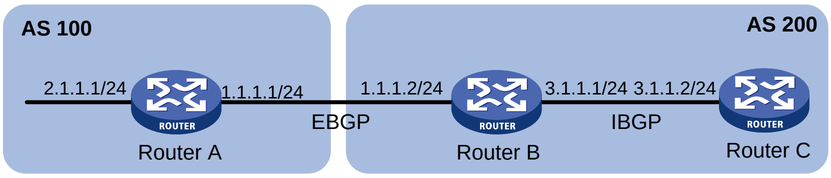
Configuring the NEXT_HOP attribute

About this task

By default, a BGP router does not set itself as the next hop for routes advertised to an IBGP peer or peer group. In some cases, however, you must configure the advertising router as the next hop to ensure that the BGP peer can find the correct next hop.

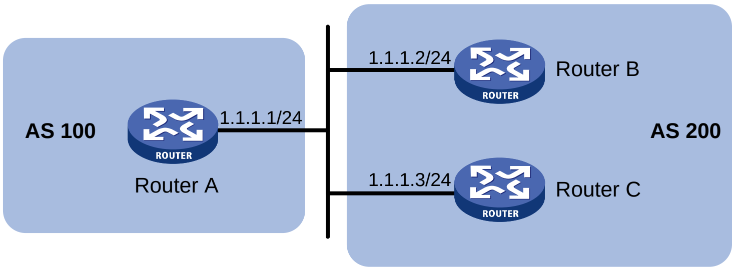
For example, as shown in Figure 12, Router A and Router B establish an EBGP neighbor relationship, and Router B and Router C establish an IBGP neighbor relationship. If Router C has no route destined for IP address 1.1.1.1/24, you must configure Router B to set itself 3.1.1.1/24 as the next hop for the network 2.1.1.1/24 advertised to Router C.

Figure 12 NEXT_HOP attribute configuration



If a BGP router has two peers on a broadcast network, it does not set itself as the next hop for routes sent to an EBGP peer by default. As shown in Figure 13, Router A and Router B establish an EBGP neighbor relationship, and Router B and Router C establish an IBGP neighbor relationship. They are on the same broadcast network 1.1.1.0/24. When Router B sends EBGP routes to Router A, it does not set itself as the next hop by default. However, you can configure Router B to set it (1.1.1.2/24) as the next hop for routes sent to Router A by using the **peer next-hop-local** command as needed.

Figure 13 NEXT_HOP attribute configuration



Restrictions and guidelines

If you have configured BGP load balancing, the router sets itself as the next hop for routes sent to an IBGP peer or peer group regardless of whether the **peer next-hop-local** command is configured.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.
   system-view
2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:
     bgp *as-number* [**instance** *instance-name*]

address-family ipv4 [**unicast**]

- Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv4 [**unicast**]
- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 multicast

3. Specify the router as the next hop for routes sent to a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **next-hop-local**

   By default, the router sets itself as the next hop for routes sent to an EBGP peer or peer group. However, it does not set itself as the next hop for routes sent to an IBGP peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast
3. Specify the router as the next hop for routes sent to a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **next-hop-local**

   By default, the router sets itself as the next hop for routes sent to an EBGP peer or peer group. However, it does not set itself as the next hop for routes sent to an IBGP peer or peer group.

Setting a preferred value for received routes

About this task

Perform this task to set a preferred value for specific routes to control BGP path selection.

Among multiple routes that have the same destination/mask and are learned from different peers, the one with the greatest preferred value is selected as the optimal route.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast
3. Set a preferred value for routes received from a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **preferred-value** *value*

   By default, the preferred value is 0 for routes received from a peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast
3. Set a preferred value for routes received from a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **preferred-value** *value*

   By default, the preferred value is 0 for routes received from a peer or peer group.

Configuring the default local preference

About this task

The local preference is used to determine the optimal route for traffic leaving the local AS. When a BGP router obtains from several IBGP peers multiple routes to the same destination, but with different next hops, it selects the route with the highest local preference as the optimal route.

This task allows you to specify the default local preference for routes sent to IBGP peers.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure the default local preference.

   default local-preference *value*

   The default local preference is 100.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

address-family ipv6 multicast

3. Configure the default local preference.

   default local-preference *value*

   The default local preference is 100.

Configuring the AS_PATH attribute

Permitting local AS number to appear in routes from a peer or peer group

About this task

In general, BGP checks whether the AS_PATH attribute of a route from a peer contains the local AS number. If yes, it discards the route to avoid routing loops.

In certain network environments, however, the AS_PATH attribute of a route from a peer must be allowed to contain the local AS number. Otherwise, the route cannot be advertised correctly.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]

   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Permit the local AS number to appear in routes from a peer or peer group and set the appearance times.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **allow-as-loop** [*number*]

   By default, the local AS number is not allowed in routes from a peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.

   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

bgp *as-number* [**instance** *instance-name*]

address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv6 multicast

3. Permit the local AS number to appear in routes from a peer or peer group and set the appearance times.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **allow-as-loop** [*number*]

   By default, the local AS number is not allowed in routes from a peer or peer group.

Ignoring the AS_PATH attribute during optimal route selection

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure BGP to ignore the AS_PATH attribute during optimal route selection.

   bestroute as-path-neglect [**all-instance**]

   By default, BGP includes the AS_PATH attribute in optimal route selection.

   The **all-instance** keyword is supported only in BGP instance view.

Advertising a fake AS number to a peer or peer group

About this task

After you move a BGP router from an AS to another AS (from AS 2 to AS 3 for example), you have to modify the AS number of the router on all its EBGP peers. To avoid such modifications, you can configure the router to advertise a fake AS number 2 to its EBGP peers so that the EBGP peers still think that Router A is in AS 2.

Restrictions and guidelines

This command applies only to EBGP peers or EBGP peer groups.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.
     ```
     bgp as-number [ instance instance-name ]
     ```
   - Execute the following commands in sequence to enter BGP-VPN instance view:
     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     ```
3. Advertise a fake AS number to a peer or peer group.
   ```
   peer { group-name | ipv4-address [ mask-length ] } fake-as as-number
   ```
   By default, no fake AS number is advertised to a peer or peer group.

Procedure (IPv6 unicast/multicast address family)

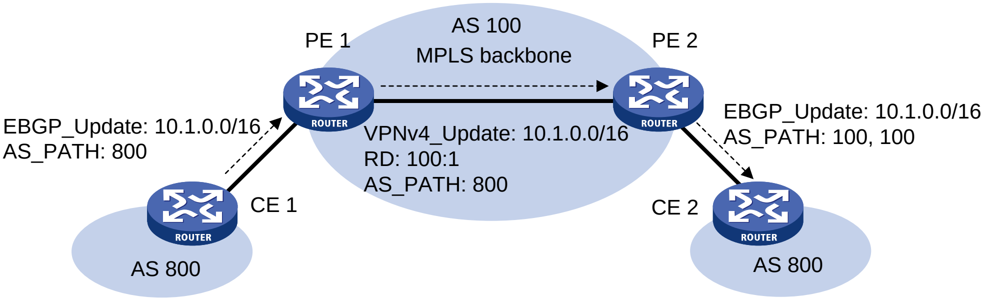
1. Enter system view.
   ```
   system-view
   ```
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.
     ```
     bgp as-number [ instance instance-name ]
     ```
   - Execute the following commands in sequence to enter BGP-VPN instance view:
     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     ```
3. Advertise a fake AS number to a peer or peer group.
   ```
   peer { group-name | ipv6-address [ prefix-length ] } fake-as as-number
   ```
   By default, no fake AS number is advertised to a peer or peer group.

Configuring AS number substitution

About this task

To use EBGP between PE and CE in MPLS L3VPN, VPN sites in different geographical areas should have different AS numbers. Otherwise, BGP discards route updates containing the local AS number. If two CEs connected to different PEs use the same AS number, you must configure AS number substitution on each PE. This substitution can replace the AS number in route updates originated by the remote CE as its own AS number before advertising them to the connected CE.

Figure 14 AS number substitution configuration (in an IPv4 network)



As shown in Figure 14, CE 1 and CE 2 use the same AS number 800. To ensure bidirectional communication between the two sites, configure AS number substitution on PE 2. PE 2 replaces AS 800 with AS 100 for the BGP route update originated from CE 1 before advertising it to CE 2. Perform the same configuration on PE 1.

Restrictions and guidelines

Do not configure AS number substitution in normal circumstances. Otherwise, routing loops might occur.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure AS number substitution for a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **substitute-as**

   By default, AS number substitution is not configured.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure AS number substitution for a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **substitute-as**

   By default, AS number substitution is not configured.

Removing private AS numbers from sent updates

About this task

Private AS numbers are typically used in test networks, and should not be transmitted in public networks. The range of private AS numbers is from 64512 to 65535.

Perform this task to enable BGP to remove private AS numbers from the AS_PATH attribute of updates sent to a peer or peer group.

Restrictions and guidelines

This feature is applicable only to EBGP peers or peer groups.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.

- Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 [**unicast**]
- Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv4 [**unicast**]
- Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv4 multicast

3. Configure BGP to remove private AS numbers from the AS_PATH attribute of updates sent to an EBGP peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **public-as-only**

   By default, BGP updates sent to an EBGP peer or peer group can carry both public and private AS numbers.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast
3. Configure BGP to remove private AS numbers from the AS_PATH attribute of updates sent to an EBGP peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **public-as-only**

   By default, BGP updates sent to an EBGP peer or peer group can carry both public and private AS numbers.

Ignoring the first AS number of EBGP route updates

About this task

By default, BGP checks the first AS number of an EBGP-learned route update. If the first AS number is neither the AS number of the BGP peer nor a private AS number, the BGP router disconnects the BGP session to the peer.

Ignoring the first AS number of all EBGP route updates

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable BGP to ignore the first AS number of EBGP route updates.

   ignore-first-as

   By default, BGP checks the first AS number of EBGP-learned route updates.

Ignoring the first AS number of EBGP route updates received from a peer or peer group (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BGP to ignore the first AS number of EBGP route updates received from a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **ignore-first-as**

   By default, BGP checks the first AS number of EBGP-learned route updates.

Ignoring the first AS number of EBGP route updates received from a peer or peer group (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BGP to ignore the first AS number of EBGP route updates received from a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ignore-first-as**

   By default, BGP checks the first AS number of EBGP-learned route updates.

Configuring the MED attribute

About the MED attribute

BGP uses MED to determine the optimal route for traffic going into an AS. When a BGP router obtains multiple routes with the same destination but with different next hops, it selects the route with the smallest MED value as the optimal route if other conditions are the same.

Configuring the default MED value

Configuring the default MED value (IPv4 unicast/multicast address family)

1. Enter system view.

   `system-view`

2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

3. Configure the default MED value.

   default med *med-value*

   The default MED value is 0.

Configuring the default MED value (IPv6 unicast/multicast address family)

1. Enter system view.

   `system-view`

2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

  ```
  bgp as-number [ instance instance-name ]
  address-family ipv6 multicast
  ```

3. Configure the default MED value.

   ```
   default med med-value
   ```

   The default MED value is 0.

Enabling MED comparison for routes from different ASs

About this task

By default, BGP only compares the MEDs of routes from the same AS. This task enables BGP to compare the MEDs of routes from different ASs.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     ```
     bgp as-number [ instance instance-name ]
     ```

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     ```

3. Enable MED comparison for routes from different ASs.

   ```
   compare-different-as-med
   ```

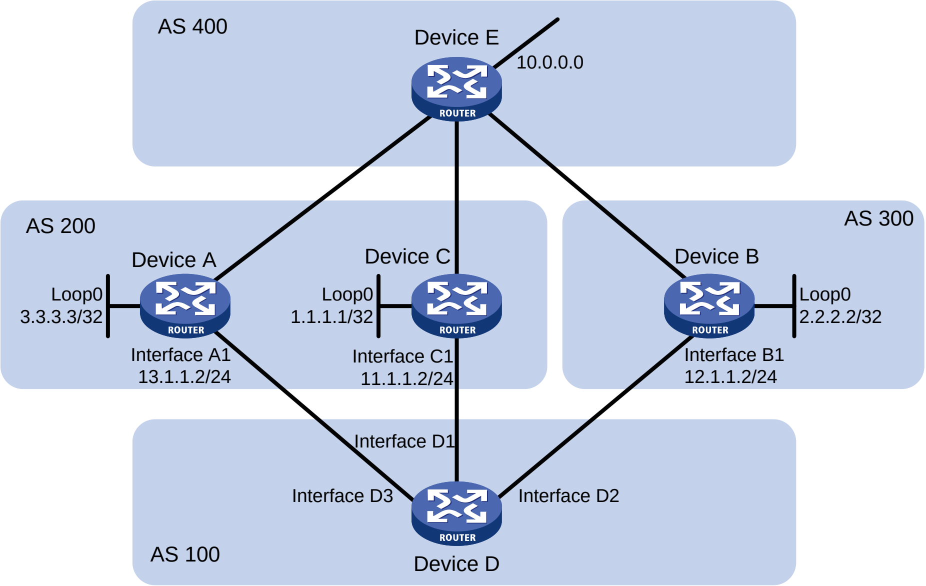
   By default, MED comparison for routes from different ASs is disabled.

Enabling MED comparison for routes on a per-AS basis

About this task

This task enables BGP to compare the MEDs of routes from an AS.

Figure 15 Route selection based on MED (in an IPv4 network)



As shown in Figure 15, Device D establishes indirect EBGP peer relationships with Device A, Device B, and Device C, and learns addresses 1.1.1.1/32, 2.2.2.2/32, and 3.3.3.3/32 through OSPF. The following output shows the routing information on Device D.

```
Destination/Mask   Proto   Pre Cost        NextHop         Interface
1.1.1.1/32         O_INTRA 10  10          11.1.1.2        Interface D1
2.2.2.2/32         O_INTRA 10  20          12.1.1.2        Interface D2
3.3.3.3/32         O_INTRA 10  30          13.1.1.2        Interface D3
```

Device D learns network 10.0.0.0 from both Device A and Device B. Because the route learned from Device B has a smaller IGP metric, the route is optimal. The following output shows the BGP routing table on Device D.

```
     Network             NextHop          MED         LocPrf      PrefVal Path/Ogn
 *>e  10.0.0.0           2.2.2.2          50                      0       300 400e
 * e                     3.3.3.3          50                      0       200 400e
```

When Device D learns network 10.0.0.0 from Device C, it compares the route with the optimal route in its routing table. Because Device C and Device B reside in different ASs, BGP does not compare the MEDs of the two routes. The route from Device C has a smaller IGP metric than the route from Device B, so the route from Device C becomes optimal. The following output shows the BGP routing table on Device D.

```
     Network             NextHop          MED         LocPrf      PrefVal Path/Ogn
 *>e  10.0.0.0           1.1.1.1          60                      0       200 400e
 * e  10.0.0.0           2.2.2.2          50                      0       300 400e
 * e                     3.3.3.3          50                      0       200 400e
```

However, Device C and Device A reside in the same AS, and Device C has a greater MED, so network 10.0.0.0 learned from Device C should not be optimal.

To avoid this problem, you can configure the **bestroute compare-med** command to enable MED comparison for routes from the same AS on Device D. After that, Device D puts the routes received from each AS into a group, selects the route with the lowest MED from each group, and compares routes from different groups. Network 10.0.0.0 learned from Device B is the optimal route. The following output shows the BGP routing table on Device D.

```
     Network             NextHop          MED         LocPrf      PrefVal Path/Ogn
 *>e  10.0.0.0           2.2.2.2          50                      0       300 400e
 * e                    3.3.3.3         50                     0       200 400e
 * e                    1.1.1.1         60                     0       200 400e
```

Procedure

1. Enter system view.

   `system-view`

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     `bgp` *`as-number`* `[` **`instance`** *`instance-name`* `]`

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     `bgp` *`as-number`* `[` **`instance`** *`instance-name`* `]`

     `ip vpn-instance` *`vpn-instance-name`*

3. Enable MED comparison for routes on a per-AS basis.

   `bestroute compare-med` `[` **`all-instance`** `]`

   By default, MED comparison for routes on a per-AS basis is disabled.

   The **`all-instance`** keyword is supported only in BGP instance view.

Enabling MED comparison for routes from confederation peers

About this task

This task enables BGP to compare the MEDs of routes received from confederation peers. However, if a route received from a confederation peer has an AS number that does not belong to the confederation, BGP does not compare the route with other routes. For example, a confederation has three AS numbers 65006, 65007, and 65009. BGP receives three routes from different confederation peers. The AS_PATH attributes of these routes are 65006 65009, 65007 65009, and 65008 65009, and the MED values of them are 2, 3, and 1. Because the third route's AS_PATH attribute contains AS number 65008 that does not belong to the confederation, BGP does not compare it with other routes. As a result, the first route becomes the optimal route.

Procedure

1. Enter system view.

   `system-view`

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     `bgp` *`as-number`* `[` **`instance`** *`instance-name`* `]`

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     `bgp` *`as-number`* `[` **`instance`** *`instance-name`* `]`

     `ip vpn-instance` *`vpn-instance-name`*

3. Enable MED comparison for routes from confederation peers.

   `bestroute med-confederation` `[` **`all-instance`** `]`

   By default, MED comparison for routes from confederation peers is disabled.

   The **`all-instance`** keyword is supported only in BGP instance view.

Ignoring IGP metrics during optimal route selection

About this task

By default, BGP includes IGP metrics in optimal route selection. If multiple routes to the same destination are available, BGP selects the route with the smallest IGP metric as the optimal route.

Perform this task to enable BGP to ignore IGP metrics during optimal route selection.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure BGP to ignore IGP metrics during optimal route selection.

   bestroute igp-metric-ignore [**all-instance**]

   By default, BGP includes IGP metrics in optimal route selection.

   The **all-instance** keyword is supported only in BGP instance view.

Ignoring router IDs during optimal route selection

About this task

By default, BGP compares router IDs during optimal route selection. If multiple routes to the same destination are available, BGP selects the route with the smallest router ID as the optimal route.

Perform this task to enable BGP to ignore router IDs during optimal route selection.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BGP to ignore router IDs during optimal route selection.

   bestroute router-id-ignore [**all-instance**]

   By default, BGP compares router IDs during optimal route selection.

   The **all-instance** keyword is supported only in BGP instance view.

Display and maintenance commands for BGP path selection control

Execute **display** commands in any view.

Displaying BGP (IPv4 unicast address family)

Task	Command
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]

Displaying BGP (IPv6 unicast address family)

Task	Command
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]

Displaying BGP (IPv4 multicast address family)

Task	Command
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]

Displaying BGP (IPv6 multicast address family)

Task	Command
Display BGP path attribute information.	**display bgp** [**instance** *instance-name*] **paths** [*as-regular-expression*]

Tuning and optimizing BGP networks

BGP network tuning and optimization tasks at a glance

To tune and optimize BGP networks, perform the following tasks:

- Establishing and resetting EBGP sessions
  - Enabling BGP to establish an EBGP session over multiple hops
  - Enabling immediate re-establishment of direct EBGP connections upon link failure
- Establishing, terminating, and resetting BGP sessions
  - Enabling 4-byte AS number suppression
  - Disabling BGP session establishment
  - Configuring BGP soft-reset
- Configuring BGP load balancing
- Protecting an EBGP peer when memory usage reaches level 2 threshold
- Flushing the suboptimal BGP route to the RIB
- Specifying a label allocation mode
- Recursing unlabeled public BGP routes to LSPs

Enabling BGP to establish an EBGP session over multiple hops

About this task

To establish an EBGP session, two routers must have a direct physical link and use directly connected interfaces. If no direct link is available, you must use the **peer ebgp-max-hop** command to enable BGP to establish an EBGP session over multiple hops and specify the maximum hops.

Restrictions and guidelines

When the BGP GTSM feature is enabled, two peers can establish an EBGP session after passing GTSM check, regardless of whether the maximum number of hops is reached.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.
   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.
     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:
     bgp *as-number* [**instance** *instance-name*]
     ip vpn-instance *vpn-instance-name*
3. Enable BGP to establish an EBGP session to an indirectly connected peer or peer group and specify the maximum hop count.

peer { *group-name* | *ipv4-address* [*mask-length*] } **ebgp-max-hop** [*hop-count*]

By default, BGP cannot establish an EBGP session to an indirectly connected peer or peer group.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Enable BGP to establish an EBGP session to an indirectly connected peer or peer group and specify the maximum hop count.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ebgp-max-hop** [*hop-count*]

   By default, BGP cannot establish an EBGP session to an indirectly connected peer or peer group.

Enabling immediate re-establishment of direct EBGP connections upon link failure

About this task

By default, when the link to a directly connected EBGP peer goes down, the router does not re-establish a session to the peer until the hold time timer expires. This feature enables BGP to immediately recreate the session in that situation. When this feature is disabled, route flapping does not affect EBGP session state.

Procedure

1. Enter system view.

   system-view
2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]
3. Enable immediate re-establishment of direct EBGP connections upon link failure.

   ebgp-interface-sensitive

   By default, immediate re-establishment of direct EBGP connections is enabled.

Enabling 4-byte AS number suppression

About this task

BGP supports 4-byte AS numbers. The 4-byte AS number occupies four bytes, in the range of 1 to 4294967295. By default, a device sends an OPEN message to the peer device for session establishment. The OPEN message indicates that the device supports 4-byte AS numbers. If the peer device supports 2-byte AS numbers instead of 4-byte AS numbers, the session cannot be established. To resolve this issue, enable the 4-byte AS number suppression feature. The device then sends an OPEN message to inform the peer that it does not support 4-byte AS numbers, so the BGP session can be established.

Restrictions and guidelines

If the peer device supports 4-byte AS numbers, do not enable the 4-byte AS number suppression feature. Otherwise, the BGP session cannot be established.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable 4-byte AS number suppression.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **capability-advertise suppress-4-byte-as**

   By default, 4-byte AS number suppression is disabled.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable 4-byte AS number suppression.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **capability-advertise suppress-4-byte-as**

   By default, 4-byte AS number suppression is disabled.

Disabling BGP session establishment

About disabling BGP session establishment

This task enables you to temporarily tear down BGP sessions to a peer/peer group. You can perform network upgrade and maintenance without needing to delete and reconfigure the peers and peer groups. To recover the sessions, execute the **undo peer ignore** command.

Disabling BGP session establishment with a peer or peer group (IPv4 unicast/multicast address family)

1. Enter system view.

system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Disable BGP session establishment with a peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **ignore**

   By default, BGP can establish a session to a peer or peer group.

CAUTION:

- If a session has been established to a peer, executing the **peer ignore** command for the peer tears down the session and clears all related routing information.
- If sessions have been established to a peer group, executing the **peer ignore** command for the peer group tears down the sessions to all peers in the group and clears all related routing information.

Disabling BGP session establishment with a peer or peer group (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*
3. Disable BGP session establishment with a peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ignore**

   By default, BGP can establish a session to a peer or peer group.

CAUTION:

- If a session has been established to a peer, executing the **peer ignore** command for the peer tears down the session and clears all related routing information.
- If sessions have been established to a peer group, executing the **peer ignore** command for the peer group tears down the sessions to all peers in the group and clears all related routing information.

Configuring BGP soft-reset

About BGP soft-reset

After you modify the route selection policy, for example, modify the preferred value, you must reset BGP sessions to apply the new policy. The reset operation tears down and re-establishes BGP sessions.

To avoid tearing down BGP sessions, you can use one of the following soft-reset methods to apply the new policy:

- **Enabling route refresh**—The BGP router advertises a ROUTE-REFRESH message to the specified peer, and the peer resends its routing information to the router. After receiving the routing information, the router filters the routing information by using the new policy.

  This method requires that both the local router and the peer support route refresh.
- **Saving updates**—Use the **`peer keep-all-routes`** command to save all route updates from the specified peer. After modifying the route selection policy, filter routing information by using the new policy.

  This method does not require that the local router and the peer support route refresh but it uses more memory resources to save routes.
- **Manual soft-reset**—Use the **`refresh bgp`** command to enable BGP to send local routing information or advertise a ROUTE-REFRESH message to the specified peer. The peer then resends its routing information. After receiving the routing information, the router filters the routing information by using the new policy.

  This method requires that both the local router and the peer support route refresh.

Enabling route refresh (IPv4 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     ```
     bgp as-number [ instance instance-name ]
     ```
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     ```
     bgp as-number [ instance instance-name ]
     ip vpn-instance vpn-instance-name
     ```
3. Enable BGP route refresh for a peer or peer group.
   - Enable BGP route refresh for the specified peer or peer group.

     ```
     peer { group-name | ipv4-address [ mask-length ] }
     capability-advertise route-refresh
     ```
   - Enable the BGP route refresh, multi-protocol extension, and 4-byte AS number features for the specified peer or peer group.

     ```
     undo peer { group-name | ipv4-address [ mask-length ] }
     capability-advertise conventional
     ```

   By default, the BGP route refresh, multi-protocol extension, and 4-byte AS number features are enabled.

Enabling route refresh (IPv6 unicast/multicast address family)

1. Enter system view.

   ```
   system-view
   ```
2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     ```
     bgp as-number [ instance instance-name ]
     ```

- Execute the following commands in sequence to enter BGP-VPN instance view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

3. Enable BGP route refresh for a peer or peer group.
   - Enable BGP route refresh for the specified peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **capability-advertise route-refresh**
   - Enable the BGP route refresh, multi-protocol extension, and 4-byte AS number features for the specified peer or peer group.

     undo peer { *group-name* | *ipv6-address* [*prefix-length*] } **capability-advertise conventional**

   By default, the BGP route refresh, multi-protocol extension, and 4-byte AS number features are enabled.

Saving updates (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast
3. Save all route updates from the peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **keep-all-routes**

   By default, route updates from peers and peer groups are not saved.

   This command takes effect only for the routes received after this command is executed.

Saving updates (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view
2. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

  address-family ipv6 [**unicast**]

- Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

  bgp *as-number* [**instance** *instance-name*]

  address-family ipv6 multicast

3. Save all route updates from the peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **keep-all-routes**

   By default, route updates from peers and peer groups are not saved.

   This command takes effect only for the routes received after this command is executed.

Configuring manual soft-reset (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BGP route refresh for a peer or peer group.
   - Enable BGP route refresh for the specified peer or peer group.

     peer { *group-name* | *ipv4-address* [*mask-length*] } **capability-advertise route-refresh**

   - Enable the BGP route refresh, multi-protocol extension, and 4-byte AS number features for the specified peer or peer group.

     undo peer { *group-name* | *ipv4-address* [*mask-length*] } **capability-advertise conventional**

   By default, the BGP route refresh, multi-protocol extension, and 4-byte AS number features are enabled.

4. Perform manual soft-reset:

   a. Return to system view.

      quit

   b. Return to user view.

      quit

   c. Perform manual soft-reset.

      refresh bgp [**instance** *instance-name*] { *ipv4-address* [*mask-length*] | **all** | **external** | **group** *group-name* | **internal** } { **export** | **import** } **ipv4** [**multicast** | [**unicast**] [**vpn-instance** *vpn-instance-name*]]

Configuring manual soft-reset (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BGP route refresh for a peer or peer group.
   - Enable BGP route refresh for the specified peer or peer group.

     peer { *group-name* | *ipv6-address* [*prefix-length*] } **capability-advertise route-refresh**
   - Enable the BGP route refresh, multi-protocol extension, and 4-byte AS number features for the specified peer or peer group.

     undo peer { *group-name* | *ipv6-address* [*prefix-length*] } **capability-advertise conventional**

   By default, the BGP route refresh, multi-protocol extension, and 4-byte AS number features are enabled.

4. Perform manual soft-reset:

   a. Return to system view.

      quit

   b. Return to user view.

      quit

   c. Perform manual soft-reset.

      refresh bgp [**instance** *instance-name*] { *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } { **export** | **import** } **ipv6** [**multicast** | [**unicast**] [**vpn-instance** *vpn-instance-name*]]

Configuring BGP load balancing

About this task

Perform this task to specify the maximum number of BGP ECMP routes for load balancing.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

ip vpn-instance *vpn-instance-name*

3. (Optional.) Enable BGP to ignore IGP metrics during optimal route selection.

   bestroute igp-metric-ignore

   By default, BGP compares IGP metrics during optimal route selection, and selects the route with the smallest IGP metric as the optimal route.

   BGP cannot use routes with different IGP metrics to implement load balancing. To resolve this issue, you can use this command.

4. Return to system view.
   - In BGP instance view:

     quit
   - In BGP-VPN instance view:

     quit

     quit

5. Enter BGP IPv4 unicast address family view, BGP-VPN IPv4 unicast address family view, or BGP IPv4 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv4 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv4 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv4 multicast

6. Specify the maximum number of BGP ECMP routes for load balancing.

   balance [**ebgp** | **eibgp** | **ibgp**] *number*

   By default, load balancing is disabled.

7. (Optional.) Enable BGP to ignore the AS_PATH attribute when it implements load balancing.

   balance as-path-neglect

   By default, BGP does not ignore the AS_PATH attribute when it implements load balancing.

8. (Optional.) Enable BGP to perform load balancing for routes that have different AS_PATH attributes of the same length.

   balance as-path-relax

   By default, BGP cannot perform load balancing for routes that have different AS_PATH attributes of the same length.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. (Optional.) Enable BGP to ignore IGP metrics during optimal route selection.

   bestroute igp-metric-ignore

   By default, BGP compares IGP metrics during optimal route selection, and selects the route with the smallest IGP metric as the optimal route.

   BGP cannot use routes with different IGP metrics to implement load balancing. To resolve this issue, you can use this command.

4. Return to system view.
   - In BGP instance view:

     quit
   - In BGP-VPN instance view:

     quit

     quit

5. Enter BGP IPv6 unicast address family view, BGP-VPN IPv6 unicast address family view, or BGP IPv6 multicast address family view.
   - Execute the following commands in sequence to enter BGP IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP-VPN IPv6 unicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

     address-family ipv6 [**unicast**]
   - Execute the following commands in sequence to enter BGP IPv6 multicast address family view:

     bgp *as-number* [**instance** *instance-name*]

     address-family ipv6 multicast

6. Specify the maximum number of BGP ECMP routes for load balancing.

   balance [**ebgp** | **eibgp** | **ibgp**] *number*

   By default, load balancing is disabled.

7. (Optional.) Enable BGP to ignore the AS_PATH attribute when it implements load balancing.

   balance as-path-neglect

   By default, BGP does not ignore the AS_PATH attribute when it implements load balancing.

8. (Optional.) Enable BGP to perform load balancing for routes that have different AS_PATH attributes of the same length.

   balance as-path-relax

   By default, BGP cannot perform load balancing for routes that have different AS_PATH attributes of the same length.

Protecting an EBGP peer when memory usage reaches level 2 threshold

About this task

Memory usage includes the following threshold levels: normal, level 1, level 2, and level 3. When the level 2 threshold is reached, BGP periodically tears down an EBGP session to release memory resources until the memory usage falls below the level 2 threshold. You can configure this feature to avoid tearing down the EBGP session to an EBGP peer when the memory usage reaches the level 2 threshold.

For more information about memory usage thresholds, see device management configuration in *Fundamentals Configuration Guide*.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure BGP to protect an EBGP peer or peer group when the memory usage reaches level 2 threshold.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **low-memory-exempt**

   By default, BGP periodically tears down an EBGP session to release memory resources when level 2 threshold is reached.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure BGP to protect an EBGP peer or peer group when the memory usage reaches level 2 threshold.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **low-memory-exempt**

   By default, BGP tears down an EBGP session to release memory resources periodically when level 2 threshold is reached.

Flushing the suboptimal BGP route to the RIB

About this task

This feature flushes the suboptimal BGP route to the RIB when the following conditions are met:

- The optimal route is generated by the **network** command or is redistributed by the **import-route** command.
- The suboptimal route is received from a BGP peer.

After the suboptimal route is flushed to the RIB on a network, BGP immediately switches traffic to the suboptimal route when the optimal route fails.

For example, the device has a static route to the subnet 1.1.1.0/24 that has a higher priority than a BGP route. BGP redistributes the static route and receives a route to 1.1.1.0/24 from a peer. After the **flush suboptimal-route** command is executed, BGP flushes the received BGP route to the RIB as the suboptimal route. When the static route fails, BGP immediately switches traffic to the suboptimal route if inter-protocol FRR is enabled. For more information about inter-protocol FRR, see "Configuring basic IP routing."

Procedure

1. Enter system view.
   system-view
2. Enter BGP view.
   bgp *as-number* [**instance** *instance-name*]
3. Flush the suboptimal BGP route to the RIB.
   flush suboptimal-route

   By default, BGP is disabled from flushing the suboptimal BGP route to the RIB, and only the optimal route is flushed to the RIB.

Specifying a label allocation mode

About this task

BGP supports the following label allocation modes:

- **Per-prefix**—Allocates a label to each route prefix.
- **Per-next-hop**—Allocates a label to each next hop. This mode is applicable when the number of labels required by the per-prefix mode exceeds the maximum number of labels supported by the device.
- **Per-VPN-instance**—Allocates a label to each VPN instance. This mode is applicable when the number of labels required by the per-next-hop mode exceeds the maximum number of labels supported by the device.

Restrictions and guidelines

When you specify the per-prefix or per-next-hop label allocation mode, you can execute the **vpn popgo** command to specify the POPGO forwarding mode on an egress PE. The egress PE will pop the label for each packet and forward the packet out of the interface corresponding to the label.

When you specify the per-VPN instance label allocation mode, do not execute the **vpn popgo** command because it is mutually exclusive with the **label-allocation-mode per-vrf** command. The egress PE will pop the label for each packet and forward the packet through the FIB table.

Procedure

1. Enter system view.
   system-view
2. Enter BGP instance view.
   bgp *as-number* [**instance** *instance-name*]
3. Specify a label allocation mode.

label-allocation-mode { **per-prefix** | **per-vrf** }

By default, BGP allocates labels on a per-next-hop basis.

△ **CAUTION:**

A change to the label allocation mode enables BGP to re-advertise all routes, which will cause temporary service interruption. Use this feature with caution.

Recursing unlabeled public BGP routes to LSPs

About this task

To perform IP forwarding on customer packets, carrier network devices must learn a large number of routes. To reduce workload and save resources on carrier network devices, configure this feature to recurse unlabeled public BGP routes to LSPs on user access devices. This feature allows the carrier network devices to forward customer packets based on labels, without the need to learn customer network routes.

After you configure this feature, unlabeled public BGP routes will be preferentially recursed to LSPs. If a route fails to be recursed to an LSP, the route will be recursed to the IP next hop.

Restrictions and guidelines

To recurse unlabeled public IPv6 BGP routes to IPv4 tunnels, you must configure the egress node to assign a non-null label to the penultimate hop.

Procedure (IPv4 unicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enter BGP IPv4 unicast address family view.

   address-family ipv4 [**unicast**]

4. Recurse unlabeled public BGP routes to LSPs.

   unicast-route recursive-lookup tunnel [**prefix-list** *ipv4-prefix-list-name*] [**tunnel-policy** *tunnel-policy-name*]

   By default, unlabeled public BGP routes are recursed to IP next hops instead of LSPs.

Procedure (IPv6 unicast)

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enter BGP IPv6 unicast address family view.

   address-family ipv6 [**unicast**]

4. Recurse unlabeled public BGP routes to LSPs.

   unicast-route recursive-lookup tunnel [**prefix-list** *ipv6-prefix-list-name*] [**tunnel-policy** *tunnel-policy-name*]

   By default, unlabeled public BGP routes are recursed to IP next hops instead of LSPs.

Display and maintenance commands for BGP network tuning and optimization

Resetting BGP sessions

△ **CAUTION:**

A reset operation tears down BGP sessions for a short period of time.

Execute **reset** commands in user view.

Task	Command
Reset BGP sessions for IPv4 unicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv4-address* [*mask-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv4** [**unicast**] [**vpn-instance** *vpn-instance-name*]
Reset BGP sessions for IPv4 multicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv4-address* [*mask-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv4 multicast**
Reset BGP sessions for IPv6 unicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv6** [**unicast**] [**vpn-instance** *vpn-instance-name*] **reset bgp** *ipv4-address* [*mask-length*] **ipv6** [**unicast**]
Reset BGP sessions for IPv6 multicast address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } **ipv6 multicast**
Reset all BGP sessions.	**reset bgp** [**instance** *instance-name*] **all**

Configuring BGP security features

BGP security feature configuration tasks at a glance

To configure BGP security features, perform the following tasks:

- Enabling MD5 authentication for BGP peers
- Enabling keychain authentication for BGP peers
- Configuring GTSM for BGP
- Configuring IPsec for IPv6 BGP

Enabling MD5 authentication for BGP peers

About this task

MD5 authentication provides the following benefits:

- Peer authentication ensures that only BGP peers that have the same password can establish TCP connections.
- Integrity check ensures that BGP packets exchanged between peers are intact.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable MD5 authentication for a BGP peer group or peer.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **password** { **cipher** | **simple** } *password*

   By default, MD5 authentication is disabled.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.

   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable MD5 authentication for a BGP peer group or peer.

peer { *group-name* | *ipv6-address* [*prefix-length*] } **password** { **cipher** | **simple** } *password*

By default, MD5 authentication is disabled.

Enabling keychain authentication for BGP peers

About this task

Keychain authentication enhances the security of BGP in the following ways:

- BGP peers can establish TCP connections only when they are both enabled with keychain authentication.
- The keys used by the BGP peers at the same time must have the same ID.
- The keys with the same ID must use the same authentication algorithm and key string.

For more information about keychains, see *Security Configuration Guide*.

Restrictions and guidelines

Follow these restrictions and guidelines when you configure the algorithm and key ID in keychain authentication:

- BGP supports the HMAC-MD5, HMAC-SHA-256, HMAC-SM3, SM3, and MD5 algorithms. To configure an algorithm, execute the **authentication-algorithm** command.
- BGP supports key IDs in the range of 0 to 63. To configure a key ID, execute the **key** command.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable keychain authentication for a BGP peer or peer group.

   peer { *group-name* | *ip-address* [*mask-length*] } **keychain** *keychain-name*

   By default, keychain authentication is disabled.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable keychain authentication for a BGP peer or peer group.

peer { *group-name* | *ipv6-address* [*prefix-length*] } **keychain** *keychain-name*

By default, keychain authentication is disabled.

Configuring GTSM for BGP

About this task

The Generalized TTL Security Mechanism (GTSM) protects a BGP session by comparing the TTL value in the IP header of incoming BGP packets against a valid TTL range. If the TTL value is within the valid TTL range, the packet is accepted. If not, the packet is discarded.

The valid TTL range is from 255 – the configured hop count + 1 to 255.

When GTSM is configured, the BGP packets sent by the device have a TTL of 255.

GTSM provides best protection for directly connected EBGP sessions, but not for multihop EBGP or IBGP sessions because the TTL of packets might be modified by intermediate devices.

Restrictions and guidelines

When GTSM is configured, the local device can establish an EBGP session to the peer after both devices pass GTSM check, regardless of whether the maximum number of hops is reached.

To use GTSM, you must configure GTSM on both the local and peer devices. You can specify different *hop-count* values for them.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure GTSM for the specified BGP peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **ttl-security hops** *hop-count*

   By default, GTSM is disabled.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]

   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Configure GTSM for the specified BGP peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ttl-security hops** *hop-count*

By default, GTSM is disabled.

Configuring IPsec for IPv6 BGP

About this task

Perform this task to configure IPsec for IPv6 BGP. IPsec can provide privacy, integrity, and authentication for IPv6 BGP packets exchanged between BGP peers.

When two IPv6 BGP peers are configured with IPsec (for example, Device A and Device B), Device A encapsulates an IPv6 BGP packet with IPsec before sending it to Device B. If Device B successfully receives and de-encapsulates the packet, it establishes an IPv6 BGP peer relationship with Device A and learns IPv6 BGP routes from Device A. If Device B receives but fails to de-encapsulate the packet, or receives a packet not protected by IPsec, it discards the packet.

Procedure

1. Enter system view.

   system-view

2. Configure an IPsec transform set and a manual IPsec profile.

   See *Security Configuration Guide*.

3. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

4. Apply the IPsec profile to an IPv6 BGP peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **ipsec-profile** *profile-name*

   By default, no IPsec profile is configured for any IPv6 BGP peer or peer group.

   This command supports only IPsec profiles in manual mode.

Improving BGP network reliability

BGP network reliability improvement tasks at a glance

To improve the BGP network reliability, perform the following tasks:

- Configuring BGP GR
- Configuring BGP NSR
- Configuring BFD for BGP
- Configuring BGP FRR

Configuring BGP GR

About this task

Graceful Restart (GR) ensures forwarding continuous when a routing protocol restarts or an active/standby switchover occurs. Two routers are required to complete a GR process. The following are router roles in a GR process:

- **GR restarter**—Performs GR upon a BGP restart or active/standby switchover.
- **GR helper**—Helps the GR restarter to complete the GR process.

A device can act as a GR restarter and GR helper at the same time.

BGP GR works as follows:

1. The BGP GR restarter and helper exchange OPEN messages for GR capability negotiation. If both parties have the GR capability, they establish a GR-capable session. The GR restarter sends the GR timer set by the **`graceful-restart timer restart`** command to the GR helper in an OPEN message.
2. When an active/standby switchover occurs or BGP restarts, the GR restarter does not remove existing BGP routes from Routing Information Base (RIB) and Forwarding Information Base (FIB). It still uses these routes for packet forwarding, and it starts the RIB purge timer (set by the **`graceful-restart timer purge-time`** command). The GR helper marks all routes learned from the GR restarter as stale instead of deleting them. It continues to use these routes for packet forwarding. During the GR process, packet forwarding is not interrupted.
3. After the active/standby switchover or BGP restart completes, the GR restarter re-establishes a BGP session to the GR helper. If the BGP session fails to be established within the GR timer advertised by the GR restarter, the GR helper removes the stale routes.
4. If the BGP session is established, routing information is exchanged for the GR restarter to retrieve route entries and for the GR helper to recover stale routes.
5. Both the GR restarter and the GR helper start the End-Of-RIB marker waiting timer.

   The End-Of-RIB marker waiting time is set by the **`graceful-restart timer wait-for-rib`** command. If routing information exchange is not completed within the time, the GR restarter does not receive new routes. The GR restarter updates the RIB with the BGP routes already learned, and removes the aged routes from the RIB. The GR helper removes the stale routes.
6. The GR restarter quits the GR process if routing information exchange is not completed within the RIB purge timer. It updates the RIB with the BGP routes already learned, and removes the aged routes.

Restrictions and guidelines

Follow these guidelines when you configure BGP GR:

- The End-Of-RIB indicates the end of route updates.
- The maximum time to wait for the End-of-RIB marker configured on the local end is not advertised to the peer. It controls the time for the local end to receive updates from the peer. Set a large value for the maximum time to wait for the End-of-RIB marker when a large number of routes need to be exchanged.
- As a best practice, perform the BGP GR configuration on both the GR restarter and GR helper.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable GR capability for BGP.

   graceful-restart

   By default, GR capability is disabled for BGP.

4. Configure the GR timer.

   graceful-restart timer restart *timer*

   The default setting is 150 seconds.

   The time that a peer waits to re-establish a session must be less than the hold time.

5. Configure the maximum time to wait for the End-of-RIB marker.

   graceful-restart timer wait-for-rib *timer*

   The default setting is 180 seconds.

6. Configure the RIB purge timer.

   graceful-restart timer purge-time *timer*

   The default setting is 480 seconds.

Configuring BGP NSR

About this task

To use BGP nonstop routing (NSR), the system must have a minimum of two IRF member devices.

NSR ensures nonstop services when BGP has redundant processes on multiple IRF member devices. In contrast to GR, NSR does not require a neighbor device to recover routing information.

BGP NSR backs up BGP state and data information from the active BGP process to the standby BGP process. The standby BGP process takes over when any of the following events occurs:

- The active BGP process restarts.
- The MPU that runs the active BGP process fails.
- An ISSU starts on the MPU that runs the active BGP process.

Restrictions and guidelines

When both GR and NSR are configured for BGP, NSR has a higher priority than GR. The device will not act as the GR restarter. If the device acts as a GR helper, it cannot help the restarter to complete GR.

Procedure

1. Enter system view.

system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enable BGP NSR.

   non-stop-routing

   By default, BGP NSR is disabled.

Configuring BFD for BGP

About this task

BGP maintains neighbor relationships based on the keepalive timer and hold timer in seconds. It requires that the hold time must be at least three times the keepalive interval. This mechanism slows down link failure detection. Once a failure occurs on a high-speed link, a large quantity of packets will be dropped before routing convergence completes. BFD for BGP can solve this problem by fast detecting link failures to reduce convergence time.

Before you enable BFD for a BGP peer or peer group, you must establish a BGP session between the local router and the peer or peer group.

For more information about BFD, see BFD configuration in *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

If you have enabled GR, use BFD with caution because BFD might detect a failure before the system performs GR, which will result in GR failure. If you have enabled both BFD and GR for BGP, do not disable BFD during a GR process to avoid GR failure.

For BGP sessions established with link-local addresses, you can use only single-hop BFD to detect the link between BGP peers.

To establish a BFD session to a BGP peer, you must configure the same BFD detection mode (multi-hop or single-hop) on the local router and the BGP peer.

Procedure (IPv4 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enable BFD to detect the link to the specified BGP peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] } **bfd** [**multi-hop** | **single-hop**]

   By default, BFD is disabled.

Procedure (IPv6 unicast/multicast address family)

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

bgp *as-number* [**instance** *instance-name*]

- Execute the following commands in sequence to enter BGP-VPN instance view:

  bgp *as-number* [**instance** *instance-name*]

  ip vpn-instance *vpn-instance-name*

3. Enable BFD to detect the link to the specified IPv6 BGP peer or peer group.

   peer { *group-name* | *ipv6-address* [*prefix-length*] } **bfd** [**multi-hop** | **single-hop**]

   By default, BFD is disabled.

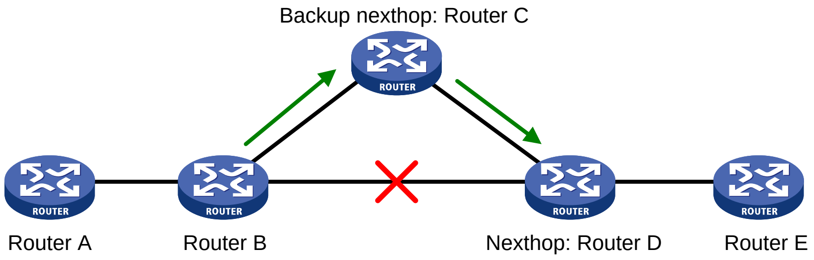
Configuring BGP FRR

About BGP FRR

When a link fails, the packets on the link are discarded, and a routing loop might occur until BGP completes routing convergence based on the new network topology.

You can enable BGP fast reroute (FRR) to resolve this issue.

Figure 16 Network diagram for BGP FRR



After you configure FRR on Router B as shown in Figure 16, BGP generates a backup next hop Router C for the primary route. BGP uses ARP or BFD echo packet mode in an IPv4 network or ND in an IPv6 network to detect the connectivity to Router D. When the link to Router D fails, BGP directs packets to the backup next hop. At the same time, BGP calculates a new optimal route, and forwards packets over the optimal route.

You can use the following methods to configure BGP FRR:

- **Method 1**—Execute the **pic** command in BGP address family view. BGP calculates a backup next hop for each BGP route in the address family if there are two or more unequal-cost routes that reach the destination.
- **Method 2**—Execute the **fast-reroute route-policy** command to use a routing policy in which a backup next hop is specified by using the command **apply** [**ipv6**] **fast-reroute backup-nexthop**. The backup next hop calculated by BGP must be the same as the specified backup next hop. Otherwise, BGP does not generate a backup next hop for the primary route. You can also configure **if-match** clauses in the routing policy to identify the routes protected by FRR.

If both methods are configured, Method 2 takes precedence over Method 1.

BGP supports FRR for IPv4 and IPv6 unicast routes, but not for IPv4 and IPv6 multicast routes.

Configuring BGP FRR by using a routing policy (IPv4 unicast address family)

1. Enter system view.

   `system-view`

2. Configure the source address of echo packets.

   `bfd echo-source-ip` *`ipv4-address`*

   By default, no source address is specified for echo packets.

   This step is required when BFD echo packet mode is used to detect the connectivity to the next hop of the primary route.

   Specify a source IP address that does not belong to any local network.

   For more information about this command, see BFD commands in *Network Management and Monitoring Command Reference*.

3. Create a routing policy and enter routing policy view.

   `route-policy` *`route-policy-name`* **`permit node`** *`node-number`*

   For more information about this command, see routing policy commands in *Layer 3—IP Routing Command Reference*.

4. Set the backup next hop for FRR.

   `apply fast-reroute backup-nexthop` *`ipv4-address`*

   By default, no backup next hop is set.

   For more information about this command, see routing policy commands in *Layer 3—IP Routing Command Reference*.

5. Return to system view.

   `quit`

6. Enter BGP instance view.

   `bgp` *`as-number`* [**`instance`** *`instance-name`*]

7. Enter BGP IPv4 unicast address family view or BGP-VPN IPv4 unicast address family view.

   - Enter BGP IPv4 unicast address family view.

     `address-family ipv4` [**`unicast`**]

   - Execute the following commands in sequence to enter BGP-VPN IPv4 unicast address family view:

     `ip vpn-instance` *`vpn-instance-name`*

     `address-family ipv4` [**`unicast`**]

8. Apply a routing policy to FRR for the address family.

   `fast-reroute route-policy` *`route-policy-name`*

   By default, no routing policy is applied.

   The **`apply fast-reroute backup-nexthop`** command and **`apply ipv6 fast-reroute backup-nexthop`** command can take effect in the applied routing policy. Other **`apply`** commands do not take effect.

Configuring BGP FRR by using a routing policy (IPv6 unicast address family)

1. Enter system view.

   `system-view`

2. Create a routing policy and enter routing policy view.

   route-policy *route-policy-name* **permit node** *node-number*

   For more information about this command, see routing policy commands in *Layer 3—IP Routing Command Reference*.

3. Set the backup next hop for FRR.

   apply ipv6 fast-reroute backup-nexthop *ipv6-address*

   By default, no backup next hop is set.

   For more information about this command, see routing policy commands in *Layer 3—IP Routing Command Reference*.

4. Return to system view.

   quit

5. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

6. Enter BGP IPv6 unicast address family view or BGP-VPN IPv6 unicast address family view.

   address-family ipv6 [**unicast**]

7. Apply a routing policy to FRR for the address family.

   fast-reroute route-policy *route-policy-name*

   By default, no routing policy is applied.

   The **apply fast-reroute backup-nexthop** and **apply ipv6 fast-reroute backup-nexthop** commands can take effect in the applied routing policy. Other **apply** commands do not take effect.

Configuring BGP FRR through PIC (IPv4 unicast address family)

Restrictions and guidelines

This feature might result in routing loops. Use it with caution.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enter BGP IPv4 unicast address family view or BGP-VPN IPv4 unicast address family view.

   address-family ipv4 [**unicast**]

4. Enable BGP FRR through PIC.

   pic

By default, BGP FRR is disabled.

Configuring BGP FRR through PIC (IPv6 unicast address family)

Restrictions and guidelines

This feature might result in routing loops. Use it with caution.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view or BGP-VPN instance view.
   - Enter BGP instance view.

     bgp *as-number* [**instance** *instance-name*]
   - Execute the following commands in sequence to enter BGP-VPN instance view:

     bgp *as-number* [**instance** *instance-name*]

     ip vpn-instance *vpn-instance-name*

3. Enter BGP IPv6 unicast address family view or BGP-VPN IPv6 unicast address family view.

   address-family ipv6 [**unicast**]

4. Enable BGP FRR through PIC.

   pic

   By default, BGP FRR is disabled.

Enabling BGP FRR to detect next hop connectivity of the primary route through BFD

About this task

By default, BGP FRR uses ARP to detect the next hop connectivity of the primary route, featuring low detection speed. If the primary link fails, service traffic might not immediately switched over to the backup path, resulting in packet loss.

With this feature configured, the device automatically creates a BFD session of the IP FRR type for detecting next hop of the primary route. Upon detecting a failure, traffic immediately switches over to the backup next hop to ensure fast convergence.

Restrictions and guidelines

This feature takes effect and automatically creates a BFD session only when a backup next hop is available for the primary route.

In the current software version, BGP does not support calculating next hops for ECMP routes. The command cannot detect next hop connectivity for an ECMP route used as the primary route.

When using echo mode BFD session to detect next hop connectivity of the primary route, you only need to configure this feature on the local router.

If another protocol (for example, OSPF and IS-IS) also uses BFD to detect next hop connectivity for the primary route, the protocol automatically creates a BFD session. If the detected link is the same as the link attached to the next hop of the BGP primary route, BGP reuses the BFD session created by the protocol, instead of creating a BFD session.

Procedure

1. Enter system view.
   system-view
2. Enter BGP instance view.
   bgp *as-number* [**instance** *instance-name*]
3. Enable BGP FRR to detect next hop connectivity of the primary route through BFD.
   primary-path-detect bfd echo
   By default, BGP FRR uses ARP to detect next hop connectivity of the primary route.

Display and maintenance commands for BGP network reliability improvement

Execute **display** commands in any view.

Displaying BGP (IPv4 unicast address family)

Task	Command
Display BGP NSR status information.	**display bgp** [**instance** *instance-name*] **non-stop-routing status**

Displaying BGP (IPv6 unicast address family)

Task	Command
Display BGP NSR status information.	**display bgp** [**instance** *instance-name*] **non-stop-routing status**

Displaying BGP (IPv4 multicast address family)

Task	Command
Display BGP NSR status information.	**display bgp** [**instance** *instance-name*] **non-stop-routing status**

Displaying BGP (IPv6 multicast address family)

Task	Command
Display BGP NSR status information.	**display bgp** [**instance** *instance-name*] **non-stop-routing status**

Configuring extended BGP features

Extended BGP feature configuration tasks at a glance

To configure extended BGP features, perform the following tasks:

- Configuring BGP LS
  - Configuring basic BGP LS
  - (Optional.) Configuring BGP LS route reflection
  - (Optional.) Specifying an AS number and a router ID for BGP LS messages
  - (Optional.) Performing manual soft-reset for BGP sessions of LS address family

Configuring BGP LS

About BGP LS

The BGP Link State (LS) feature implements inter-domain and inter-AS advertisement of link state database (LSDB) and TE database (TEDB) information.

The device sends the collected link state information to the controller, which implements end-to-end traffic management and scheduling and meets the requirements of intended applications.

Configuring basic BGP LS

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Specify an AS number for an LS peer or peer group.

   peer { *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **as-number** *as-number*

   By default, no AS number is specified.

4. Create the BGP LS address family and enter its view.

   address-family link-state

5. Enable the device to exchange LS information with the peer or peer group.

   peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **enable**

   By default, the device cannot exchange LS information with the peer or peer group.

Configuring BGP LS route reflection

About this task

Perform this task to configure a BGP route reflector and its clients. The route reflector and its clients automatically form a cluster identified by the router ID of the route reflector. The route reflector forwards route updates among its clients.

Procedure

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enter BGP LS address family view.

   address-family link-state

4. Configure BGP LS route reflection.

   - Configure the device as a route reflector and specify a peer or peer group as its client.

     peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **reflect-client**

     By default, no route reflector or client is configured.

   - (Optional.) Enable route reflection between clients.

     reflect between-clients

     By default, route reflection between clients is enabled.

     This command can reduce the number of IBGP connections in an AS.

   - (Optional.) Configure the cluster ID of the route reflector.

     reflector cluster-id { *cluster-id* | *ipv4-address* }

     By default, a route reflector uses its own router ID as the cluster ID.

Specifying an AS number and a router ID for BGP LS messages

1. Enter system view.

   system-view

2. Enter BGP instance view.

   bgp *as-number* [**instance** *instance-name*]

3. Enter BGP LS address family view.

   address-family link-state

4. Specify an AS number and a router ID for BGP LS messages.

   domain-distinguisher *as-number:router-id*

   By default, the AS number and router ID of the current BGP process are used.

   Configure this command to ensure that LS messages sent by devices in the same AS have the same AS number and router ID.

Performing manual soft-reset for BGP sessions of LS address family

1. Enter system view.
   system-view
2. Enter BGP instance view.
   bgp *as-number* [**instance** *instance-name*]
3. Enable BGP route refresh.
   - Enable BGP route refresh for a peer or peer group.
     peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **capability-advertise route-refresh**
   - Enable the BGP route refresh, multi-protocol extension, and 4-byte AS number features for a peer or peer group.
     undo peer { *group-name* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **capability-advertise conventional**

   By default, the BGP route refresh, multi-protocol extension, and 4-byte AS number features are enabled.
4. Perform manual soft-reset for BGP sessions of LS address family:
   a. Return to system view.
      quit
   b. Return to user view.
      quit
   c. Perform manual soft-reset for BGP sessions of LS address family.
      refresh bgp [**instance** *instance-name*] { *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } { **export** | **import** } **link-state**

Display and maintenance commands for extended BGP features

Displaying BGP

Execute **display** commands in any view.

Displaying BGP (LS address family)

Task	Command
Display BGP LS peer group information.	**display bgp** [**instance** *instance-name*] **group link-state** [**group-name** *group-name*]
Display BGP LS information.	**display bgp** [**instance** *instance-name*] **link-state** [*ls-prefix* | **peer** { *ipv4-address* | *ipv6-address* } { **advertised** | **received** } [**statistics**] | **statistics**]
Display BGP LS peer or peer group information.	**display bgp** [**instance** *instance-name*] **peer link-state**

Task	Command
	[*ipv4-address* *mask-length* | *ipv6-address* *prefix-length* | { *ipv4-address* | *ipv6-address* | **group-name** *group-name* } **log-info** | [*ipv4-address* | *ipv6-address*] **verbose**]
Display BGP LS address family update group information.	**display bgp** [**instance** *instance-name*] **update-group link-state** [*ipv4-address* | *ipv6-address*]

Resetting BGP sessions

CAUTION:

A reset operation tears down BGP sessions for a short period of time.

Execute **reset** commands in user view.

Task	Command
Reset BGP sessions for LS address family.	**reset bgp** [**instance** *instance-name*] { *as-number* | *ipv4-address* [*mask-length*] | *ipv6-address* [*prefix-length*] | **all** | **external** | **group** *group-name* | **internal** } **link-state**

Contents

Configuring PBR

About PBR

Policy-based routing (PBR) uses user-defined policies to route packets. A policy can specify parameters for packets that match specific criteria such as ACLs or that have specific lengths. The parameters include the next hop, output interface, default next hop, and default output interface.

Packet forwarding process

When the device receives a packet, the device searches the PBR policy for a matching node to forward that packet.

- If a matching node is found and its match mode is **permit**, the device performs the following operations:
  - **a.** Uses the next hops or output interfaces specified on the node to forward the packet.
  - **b.** Searches the routing table for a route (except the default route) to forward the packet if one of the following conditions exists:
    - – No next hops or output interfaces are specified on the node.
    - – Forwarding failed based on the next hops or output interfaces.
  - **c.** Uses the default next hop or default output interface specified on the node to forward the packet if one of the following conditions exists:
    - – No matching route was found in the routing table.
    - – The routing table-based forwarding failed.
  - **d.** Uses the default route to forward the packet if one of the following conditions exists:
    - – No default next hops or default output interfaces are specified on the node.
    - – The forwarding failed based on the default next hops or default output interfaces.
- The device perfoms routing table lookup to forward the packet in either of the following conditions:
  - o No matching node is found.
  - o A matching node is found, but its match mode is **deny**.

PBR types

PBR includes the following types:

- **Local PBR**—Guides the forwarding of locally generated packets, such as ICMP packets generated by using the **`ping`** command.
- **Interface PBR**—Guides the forwarding of packets received on an interface.

Policy

A policy includes match criteria and actions to be taken on the matching packets. A policy can have one or multiple nodes as follows:

- Each node is identified by a node number. A smaller node number has a higher priority.
- A node contains **`if-match`** and **`apply`** clauses. An **`if-match`** clause specifies a match criterion, and an **`apply`** clause specifies an action.

- A node has a match mode of **permit** or **deny**.

A policy compares packets with nodes in priority order. If a packet matches the criteria on a node, it is processed by the action on the node. If the packet does not match any criteria on the node, it goes to the next node for a match. If the packet does not match the criteria on any node, the device performs a routing table lookup.

Relationship between if-match clauses

On a node, you can specify multiple types of **if-match** clauses but only one **if-match** clause for each type.

To match a node, a packet must match all types of the **if-match** clauses for the node but only one **if-match** clause for each type.

Relationship between apply clauses

You can specify multiple **apply** clauses for a node, but some of them might not be executed. For more information about relationship between **apply** clauses, see "Configuring actions for a node."

Relationship between the match mode and clauses on the node

Does a packet match all the if-match clauses on the node?	Match mode	
	Permit	**Deny**
Yes.	• If the node contains **apply** clauses, PBR executes the **apply** clauses on the node. ○ If PBR-based forwarding succeeds, PBR does not compare the packet with the next node. ○ If PBR-based forwarding fails and the **apply continue** clause is not configured, PBR does not compare the packet with the next node. ○ If PBR-based forwarding fails and the **apply continue** clause is configured, PBR compares the packet with the next node. • If the node does not contain **apply** clauses, the device performs a routing table lookup for the packet.	The device performs a routing table lookup for the packet.
No.	PBR compares the packet with the next node.	PBR compares the packet with the next node.

NOTE:

A node that has no **if-match** clauses matches any packet.

PBR and Track

PBR can work with the Track feature to dynamically adapt the availability status of an **apply** clause to the link status of a tracked object. The tracked object can be a next hop, output interface, default next hop, or default output interface.

- When the track entry associated with an object changes to **Negative**, the **apply** clause is invalid.
- When the track entry changes to **Positive** or **NotReady**, the **apply** clause is valid.

For more information about Track-PBR collaboration, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines: PBR configuration

If the device performs forwarding in software, PBR does not process IP packets destined for the local device.

If the device performs forwarding in hardware and a packet destined for it matches a PBR policy, PBR will execute the apply clauses in the policy, including the clause for forwarding. When you configure a PBR policy, be careful to avoid this situation.

PBR tasks at a glance

To configure PBR, perform the following tasks:

1. Configuring a policy
   a. Creating a node
   b. Setting match criteria for a node
   c. Configuring actions for a node
2. Specifying a policy for PBR

   Choose the following tasks as needed:
   - Specifying a policy for local PBR
   - Specifying a policy for interface PBR

Configuring a policy

Creating a node

1. Enter system view.

   system-view
2. Create a node for a policy, and enter its view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

Setting match criteria for a node

1. Enter system view.

   system-view
2. Enter policy node view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*
3. Set match criteria.
   - Set an ACL match criterion.

     if-match acl { *acl-number* | **name** *acl-name* }

     By default, no ACL match criterion is set.

The ACL match criterion cannot match Layer 2 information.

- Set a packet length match criterion.

  if-match packet-length *min-len max-len*

  By default, no packet length match criterion is set.

- Set a source IP address match criterion for local PBR.

  if-match source-ip { **interface** *interface-type interface-number* | [**vpn-instance** *vpn-instance-name*] *ip-address* }

  By default, no source IP address match criterion is set for local PBR.

  Support for this command depends on the device model. For more information, see the command reference.

- Set application group match criteria.

  if-match app-group *app-group-name*&<1-6>

  By default, no application group match criteria are set.

  Application group match criteria apply only to interface PBR.

  For more information about application groups, see APR configuration in *Security Configuration Guide*.

  Support for this command depends on the device model. For more information, see the command reference.

- Set service object group match criteria.

  if-match object-group service *object-group-name*&<1-6>

  By default, no service object group match criteria are set.

  For more information about service object groups, see object group configuration in *Security Configuration Guide*.

  Support for this command depends on the device model. For more information, see the command reference.

Configuring actions for a node

About this task

The **apply** clauses allow you to specify the actions to be taken on matching packets on a node.

The following **apply** clauses determine the packet forwarding paths in a descending order:

- **apply access-vpn**
- **apply next-hop**
- **apply output-interface**
- **apply default-next-hop**
- **apply default-output-interface**

PBR supports the **apply** clauses in Table 1.

Table 1 Apply clauses supported in PBR

Clause	Meaning	Remarks
apply precedence	Sets an IP precedence.	This clause is always executed.
apply ip-df *df-value*	Sets the Don't Fragment (DF) bit in the IP header.	This clause is always executed.

apply loadshare { **next-hop** | **output-interface** | **default-next-hop** | **default-output-interface** }	Enables load sharing among multiple next hops, output interfaces, default next hops, or default output interfaces.	Multiple next hop, output interface, default next hop, or default output interface options operate in either primary/backup or load sharing mode. • **Primary/backup mode**—One option is selected from all options in configuration order for packet forwarding, with all remaining options as backups. For example, if multiple output interfaces are configured, the first configured output interface is selected. When the selected output interface fails, the next available output interface takes over. • **Load sharing mode**—Matching traffic is distributed across the available options, as follows: ○ **Multiple output interface, default next hop, or default output interface options**—Load share traffic in round robin manner, starting from the first configured option. They perform per-packet load sharing for traffic that does not match any fast forwarding entry, and perform per-flow load sharing for traffic that matches a fast forwarding entry. ○ **Multiple next hops**—Load share traffic in proportion to their weight. By default, all next hops have the same weight and traffic is evenly distributed among them. By default, the primary/backup mode applies. For the load sharing mode to take effect, make sure multiple next hops, output interfaces, default next hops, or default output interfaces are set in the policy.
apply access-vpn	Specifies the forwarding tables that can be used for the matching packets.	Use this clause only in special scenarios that require sending packets received from one network to another network, for example, from a VPN to the public network, or from one VPN to another VPN. If a packet matches the forwarding table for a specified VPN instance, it is forwarded in the VPN.
apply remark-vpn	Enables VPN remark action.	VPN remark action marks the matching packets as belonging to the VPN instance to which they are forwarded based on the **apply access-vpn vpn-instance** command. All subsequent service modules of PBR handle the packets as belonging to the re-marked VPN instance. If the VPN remark action is not enabled, the forwarded matching packets are marked as belonging to the VPN instance or the public network from which they were received. VPN remark action applies only to packets

		that have been successfully forwarded based on the **apply access-vpn vpn-instance** command.
apply next-hop and **apply output-interface**	Sets next hops and sets output interfaces.	If both clauses are configured, only the **apply next-hop** clause is executed.
apply default-next-hop and **apply default-output-interface**	Sets default next hops and sets default output interfaces.	If both clauses are configured, only the **apply default-next-hop** clause is executed. The clauses take effect only in the following cases: • No next hops or output interfaces are set or the next hops and output interfaces are invalid. • The packet does not match any route in the routing table.
apply continue	Compares packets with the next node upon failure on the current node.	The **apply continue** clause applies when either of the the following conditions exist: • None of the following clauses is configured for packet forwarding: ○ **apply access-vpn vpn-instance** ○ **apply next-hop** ○ **apply output-interface** ○ **apply default-next-hop** ○ **apply default-output-interface** • A clause listed above is configured, but it has become invalid. Then, a routing table lookup also fails for the matching packet. **NOTE:** A clause might become invalid because the specified next hop is unreachable, packets cannot be forwarded in the specified VPN instance, or the specified output interface is down.

Restrictions and guidelines

For outbound PBR, you can specify only one next hop and the next hop must be directly connected.

If you specify a next hop or default next hop, PBR periodically performs a lookup in the FIB table to determine its availability. Temporary service interruption might occur if PBR does not update the route immediately after its availability status changes.

Configuring actions to modify packet fields

1. Enter system view.

   system-view

2. Enter policy node view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Configure actions.

   - Set an IP precedence.

apply precedence { *type* | *value* }

By default, no IP precedence is specified.

- Set the DF bit in the IP header.

  apply ip-df *df-value*

  By default, the DF bit in the IP header is not set.

Configuring actions to direct packet forwarding

1. Enter system view.

   system-view

2. Enter policy node view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Configure actions.
   - Specify the forwarding tables that can be used for the matching packets.

     apply access-vpn { **public** | **vpn-instance** *vpn-instance-name*&<1-4> }

     By default, the device forwards matching packets by using the forwarding table for the network from which the packets are received.

     You can repeat this command to specify the forwarding tables for the public network and VPN instances. The device forwards the matching packets by using the first available forwarding table selected in the order in which they are specified.

   - Enable VPN remark action to mark the matching packets as belonging to the VPN instance to which they are forwarded based on the **apply access-vpn vpn-instance** command.

     apply remark-vpn

     By default, VPN remark action is not configured.

   - Set next hops.

     apply next-hop [**vpn-instance** *vpn-instance-name* | **inbound-vpn**] { *ip-address* [**direct**] [**track** *track-entry-number*] [**weight** *weight-value*] }&<1-4>

     By default, no next hops are specified.

     On a node, you can specify a maximum of four next hops for backup or load sharing in one command line or by executing this command multiple times.

     If multiple next hops on the same subnet are specified for backup, the device first uses the subnet route for the next hops to forward packets when the primary next hop fails. If the subnet route is not available, the device selects a backup next hop.

     The value for n varies by device model. For more information, see the command reference.

   - Enable load sharing among multiple next hops.

     apply loadshare next-hop

     By default, the next hops operate in primary/backup mode.

   - Set output interfaces.

     apply output-interface { *interface-type interface-number* [**track** *track-entry-number*] }&<1-4>

     By default, no output interfaces are specified.

     On a node, you can specify a maximum of four output interfaces for backup or load sharing in one command line or by executing this command multiple times.

   - Enable load sharing among multiple output interfaces.

     apply loadshare output-interface

     By default, the output interfaces operate in primary/backup mode.

- Set default next hops.

  apply default-next-hop [**vpn-instance** *vpn-instance-name* | **inbound-vpn**] { *ip-address* [**direct**] [**track** *track-entry-number*] }&<1-4>

  By default, no default next hops are specified.

  On a node, you can specify a maximum of four default next hops for backup or load sharing in one command line or by executing this command multiple times.

  The value for n varies by device model. For more information, see the command reference.

- Enable load sharing among multiple default next hops.

  apply loadshare default-next-hop

  By default, the default next hops operate in primary/backup mode.

- Set default output interfaces.

  apply default-output-interface { *interface-type interface-number* [**track** *track-entry-number*] }&<1-4>

  By default, no default output interfaces are specified.

  On a node, you can specify a maximum of four default output interfaces for backup or load sharing in one command line or by executing this command multiple times.

- Enable load sharing among multiple default output interfaces.

  apply loadshare default-output-interface

  By default, the default output interfaces operate in primary/backup mode.

Comparing packets with the next node upon match failure on the current node

1. Enter system view.

   system-view

2. Enter policy node view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Compare packets with the next node upon match failure on the current node.

   apply continue

   By default, PBR does not compare packets with the next node upon match failure on the current node.

   This command takes effect only when the match mode of the node is **permit**.

Specifying a policy for PBR

Specifying a policy for local PBR

About this task

Perform this task to specify a policy for local PBR to guide the forwarding of locally generated packets.

Restrictions and guidelines

You can specify only one policy for local PBR and must make sure the specified policy already exists. Before you apply a new policy, you must first remove the current policy.

Local PBR might affect local services such as ping and Telnet. When you use local PBR, make sure you fully understand its impact on local services of the device.

Procedure

1. Enter system view.

```
system-view
```

2. Specify a policy for local PBR.

   ```
   ip local policy-based-route policy-name
   ```

   By default, local PBR is not enabled.

Specifying a policy for interface PBR

About this task

Perform this task to apply a policy to an interface to guide the forwarding of packets received on the interface.

Restrictions and guidelines

You can apply only one policy to an interface and must make sure the specified policy already exists. Before you can apply a new interface PBR policy to an interface, you must first remove the current policy from the interface.

You can apply a policy to multiple interfaces.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Specify a policy for interface PBR.

   ```
   ip policy-based-route policy-name
   ```

   By default, no policy is applied to an interface.

Display and maintenance commands for PBR

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display PBR policy information.	**display ip policy-based-route** [**policy** *policy-name*]
Display interface PBR configuration and statistics.	**display ip policy-based-route interface** *interface-type interface-number* [**slot** *slot-number*]

Task	Command
Display local PBR configuration and statistics.	**display ip policy-based-route local** [**slot** *slot-number*]
Display PBR configuration.	**display ip policy-based-route setup**
Clear PBR statistics.	**reset ip policy-based-route statistics** [**policy** *policy-name*]

Contents

Configuring IPv6 PBR

About IPv6 PBR

IPv6 policy-based routing (PBR) uses user-defined policies to route IPv6 packets. A policy can specify parameters for packets that match specific criteria such as ACLs or that have specific lengths. The parameters include the next hop, output interface, default next hop, and default output interface.

IPv6 packet forwarding process

When the device receives an IPv6 packet, the device searches the IPv6 PBR policy for a matching node to forward that packet.

- If a matching node is found and its match mode is **permit**, the device performs the following operations:
  - **a.** Uses the next hops or output interfaces specified on the node to forward the packet.
  - **b.** Searches the routing table for a route (except the default route) to forward the packet if one of the following conditions exists:
    - No next hops or output interfaces are specified on the node.
    - Forwarding failed based on the next hops or output interfaces.
  - **c.** Uses the default next hop or default output interface specified on the node to forward the packet if one of the following conditions exists:
    - No matching route was found in the routing table.
    - The routing table-based forwarding failed.
  - **d.** Uses the default route to forward the packet if one of the following conditions exists:
    - No default next hops or default output interfaces are specified on the node.
    - The forwarding failed based on the default next hops or default output interfaces.
- The device perfoms routing table lookup to forward the packet in either of the following conditions:
  - No matching node is found.
  - A matching node is found, but its match mode is **deny**.

IPv6 PBR types

IPv6 PBR includes the following types:

- **Local PBR**—Guides the forwarding of locally generated packets, such as the ICMP packets generated by using the **`ping`** command.
- **Interface PBR**—Guides the forwarding of packets received on an interface only.

Policy

An IPv6 policy includes match criteria and actions to be taken on the matching packets. A policy can have one or multiple nodes as follows:

- Each node is identified by a node number. A smaller node number has a higher priority.
- A node contains **`if-match`** and **`apply`** clauses. An **`if-match`** clause specifies a match criterion, and an **`apply`** clause specifies an action.

- A node has a match mode of **permit** or **deny**.

An IPv6 policy compares packets with nodes in priority order. If a packet matches the criteria on a node, it is processed by the action on the node. If the packet does not match any criteria on the node, it goes to the next node for a match. If the packet does not match the criteria on any node, the device performs a routing table lookup for the packet.

Relationship between if-match clauses

On a node, you can specify multiple **`if-match`** clauses, but only one **`if-match`** clause for each type. A packet that matches all the **`if-match`** clauses of a node matches the node.

Relationship between apply clauses

You can specify multiple **`apply`** clauses for a node, but some of them might not be executed. For more information about the relationship between the **`apply`** clauses, see "Configuring actions for an IPv6 node."

Relationship between the match mode and clauses on the node

Does a packet match all the if-match clauses on the node?	Match mode	
	In permit mode	**In deny mode**
Yes	• If the node contains **`apply`** clauses, IPv6 PBR executes the **`apply`** clauses on the node. ○ If IPv6 PBR-based forwarding succeeds, IPv6 PBR does not compare the packet with the next node. ○ If IPv6 PBR-based forwarding fails and the **`apply continue`** clause is not configured, IPv6 PBR does not compare the packet with the next node. ○ If IPv6 PBR-based forwarding fails and the **`apply continue`** clause is configured, IPv6 PBR compares the packet with the next node. • If the node does not contain **`apply`** clauses, the device performs a routing table lookup for the packet.	The device performs a routing table lookup for the packet.
No	IPv6 PBR compares the packet with the next node.	IPv6 PBR compares the packet with the next node.

NOTE:

A node that has no **`if-match`** clauses matches any packet.

IPv6 PBR and Track

IPv6 PBR can work with the Track feature to dynamically adapt the availability status of an **apply** clause to the link status of a tracked object. The tracked object can be a next hop, output interface, default next hop, or default output interface.

- When the track entry associated with an object changes to **Negative**, the **apply** clause is invalid.
- When the track entry changes to **Positive** or **NotReady**, the **apply** clause is valid.

For more information about Track and IPv6 PBR collaboration, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines: IPv6 PBR configuration

If the device performs forwarding in software, IPv6 PBR does not process IP packets destined for the local device.

If the device performs forwarding in hardware and a packet destined for it matches an IPv6 PBR policy, IPv6 PBR will execute the apply clauses in the policy, including the clause for forwarding. When you configure an IPv6 PBR policy, be careful to avoid this situation.

IPv6 PBR tasks at a glance

To configure IPv6 PBR, perform the following tasks:

1. Configuring an IPv6 policy
   a. Creating an IPv6 node
   b. Setting match criteria for an IPv6 node
   c. Configuring actions for an IPv6 node
2. Specifying a policy for IPv6 PBR

   Choose the following tasks as needed:
   - Specifying an IPv6 policy for IPv6 local PBR
   - Specifying an IPv6 policy for IPv6 interface PBR

Configuring an IPv6 policy

Creating an IPv6 node

1. Enter system view.

   system-view
2. Create an IPv6 policy or policy node and enter its view.

   ipv6 policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

Setting match criteria for an IPv6 node

1. Enter system view.

   system-view

2. Enter IPv6 policy node view.

   ```
   ipv6 policy-based-route policy-name [ deny | permit ] node node-number
   ```

3. Set match criteria.
   - Set an ACL match criterion.

     ```
     if-match acl { ipv6-acl-number | name ipv6-acl-name }
     ```

     By default, no ACL match criterion is set.

     The ACL match criterion cannot match Layer 2 information.
   - Set a packet length match criterion.

     ```
     if-match packet-length min-len max-len
     ```

     By default, no packet length match criterion is set.

Configuring actions for an IPv6 node

About this task

The **apply** clauses allow you to specify actions to take on matching packets on a node.

The following **apply** clauses determine the packet forwarding paths in a descending order:

- **apply access-vpn**
- **apply next-hop**
- **apply output-interface**
- **apply default-next-hop**
- **apply default-output-interface**

IPv6 PBR supports the **apply** clauses in Table 1.

Table 1 Apply clauses supported in IPv6 PBR

Clause	Meaning	Remarks
apply precedence	Sets an IP precedence.	This clause is always executed.
apply loadshare { **next-hop** | **output-interface** | **default-next-hop** | **default-output-interface** }	Enables load sharing among multiple next hops, output interfaces, default next hops, or default output interfaces.	Multiple next hop, output interface, default next hop, or default output interface options operate in either primary/backup or load sharing mode. • **Primary/backup mode**—One option is selected from all options in configuration order for packet forwarding, with all remaining options as backups. For example, if multiple output interfaces are configured, the first configured output interface is selected. When the selected output interface fails, the next available output interface takes over. • **Load sharing mode**—Matching traffic is distributed across the available options, as follows: ○ **Multiple output interface, default next hop, or default output interface options**—Load share traffic in round robin manner, starting from the first configured option. They perform per-packet

		load sharing for traffic that does not match any fast forwarding entry, and perform per-flow load sharing for traffic that matches a fast forwarding entry. ○ **Multiple next hops**—Load share traffic in proportion to their weight. By default, all next hops have the same weight and traffic is evenly distributed among them. By default, the primary/backup mode applies. For the load sharing mode to take effect, make sure multiple next hops, output interfaces, default next hops, or default output interfaces are set in the policy.
`apply access-vpn`	Specifies the forwarding tables that can be used for the matching packets.	Use this clause only in special scenarios that require sending packets received from one network to another network, for example, from a VPN to the public network, or from one VPN to another VPN. If a packet matches the forwarding table for a specified VPN instance, it is forwarded in the VPN.
`apply remark-vpn`	Enables VPN remark action.	VPN remark action marks the matching packets as belonging to the VPN instance to which they are forwarded based on the **`apply access-vpn vpn-instance`** command. All subsequent service modules of IPv6 PBR handle the packets as belonging to the re-marked VPN instance. If the VPN remark action is not enabled, the forwarded matching packets are marked as belonging to the VPN instance or the public network from which they were received. VPN remark action applies only to packets that have been successfully forwarded based on the **`apply access-vpn vpn-instance`** command.
`apply next-hop` and **`apply output-interface`**	Sets next hops and sets output interfaces.	If both clauses are configured, only the **`apply next-hop`** clause is executed.
`apply default-next-hop` and **`apply default-output-interface`**	Sets default next hops and sets default output interfaces.	If both clauses are configured, only the **`apply default-next-hop`** clause is executed. The clauses take effect only in the following cases: • No next hops or output interfaces are set or the next hops and output interfaces are invalid. • The IPv6 packet does not match any route in the routing table.
`apply continue`	Compares packets with the next node upon failure on the current node.	The **`apply continue`** clause applies when either of the the following conditions exist: • None of the following clauses is

		configured for packet forwarding: ○ **apply access-vpn vpn-instance** ○ **apply next-hop** ○ **apply output-interface** ○ **apply default-next-hop** ○ **apply default-output-interface** • A clause listed above is configured, but it has become invalid. Then, a routing table lookup also fails for the matching packet. **NOTE:** A clause might become invalid because the specified next hop is unreachable, packets cannot be forwarded in the specified VPN instance, or the specified output interface is down.

Restrictions and guidelines for action configuration

If you specify a next hop or default next hop, IPv6 PBR periodically performs a lookup in the FIB table to determine its availability. Temporary service interruption might occur if IPv6 PBR does not update the route immediately after its availability status changes.

Setting an IP precedence

1. Enter system view.

   system-view

2. Enter IPv6 policy node view.

   ipv6 policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Set an IP precedence.

   apply precedence { *type* | *value* }

   By default, no IP precedence is specified.

Configuring actions to direct packet forwarding

1. Enter system view.

   system-view

2. Enter IPv6 policy node view.

   ipv6 policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Configure actions for a node.
   - Specify the forwarding tables that can be used for the matching packets.

     apply access-vpn { **public** | **vpn-instance** *vpn-instance-name*&<1-4> }

     By default, the device forwards matching packets by using the forwarding table for the network from which the packets are received.

     You can repeat this command to specify the forwarding tables for the public network and VPN instances. The device forwards the matching packets by using the first available forwarding table selected in the order in which they are specified.
   - Enable VPN remark action to mark the matching packets as belonging to the VPN instance to which they are forwarded based on the **apply access-vpn vpn-instance** command.

     apply remark-vpn

By default, VPN remark action is not configured.

- Set next hops for permitted IPv6 packets.

  apply next-hop [**vpn-instance** *vpn-instance-name* | **inbound-vpn**] { *ipv6-address* [**direct**] [**track** *track-entry-number*] [**weight** *weight-value*] } &<1-4>

  By default, no next hops are specified.

  You can specify multiple next hops for backup or load sharing in one command line or by executing this command multiple times. You can specify a maximum of four next hops for a node.

  If multiple next hops on the same subnet are specified for backup, the device first uses the subnet route for the next hops to forward packets when the primary next hop fails. If the subnet route is not available, the device selects a backup next hop.

  The value for n varies by device model. For more information, see the command reference.

- Enable load sharing among multiple next hops.

  apply loadshare next-hop

  By default, the next hops operate in primary/backup mode.

- Set output interfaces.

  apply output-interface { *interface-type interface-number* [**track** *track-entry-number*] }&<1-4>

  By default, no output interfaces are specified.

  You can specify multiple output interfaces for backup or load sharing in one command line or by executing this command multiple times. You can specify a maximum of four output interfaces for a node.

- Enable load sharing among multiple output interfaces.

  apply loadshare output-interface

  By default, the output interfaces operate in primary/backup mode.

- Set default next hops.

  apply default-next-hop [**vpn-instance** *vpn-instance-name* | **inbound-vpn**] { *ipv6-address* [**direct**] [**track** *track-entry-number*] }&<1-4>

  By default, no default next hops are specified.

  You can specify multiple default next hops for backup or load sharing in one command line or by executing this command multiple times. You can specify a maximum of four default next hops for a node.

  The value for n varies by device model. For more information, see the command reference.

- Enable load sharing among multiple default next hops.

  apply loadshare default-next-hop

  By default, the default next hops operate in primary/backup mode.

- Set default output interfaces.

  apply default-output-interface { *interface-type interface-number* [**track** *track-entry-number*] }&<1-4>

  By default, no default output interfaces are specified.

  You can specify multiple default output interfaces for backup or load sharing in one command line or by executing this command multiple times. You can specify a maximum of four default output interfaces for a node.

- Enable load sharing among multiple default output interfaces.

  apply loadshare default-output-interface

  By default, the default output interfaces operate in primary/backup mode.

Comparing packets with the next node upon match failure on the current node

1. Enter system view.

   `system-view`

2. Enter IPv6 policy node view.

   ipv6 policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Compare packets with the next node upon match failure on the current node.

   `apply continue`

   By default, IPv6 PBR does not compare packets with the next node upon match failure on the current node.

   This command takes effect only when the match mode of the node is **permit**.

Specifying a policy for IPv6 PBR

Specifying an IPv6 policy for IPv6 local PBR

About this task

Perform this task to specify an IPv6 policy for IPv6 local PBR to guide the forwarding of locally generated packets.

Restrictions and guidelines

You can specify only one policy for IPv6 local PBR and must make sure the specified policy already exists. Before you apply a new policy, you must first remove the current policy.

IPv6 local PBR might affect local services, such as ping and Telnet. When you use IPv6 local PBR, make sure you fully understand its impact on local services of the device.

Procedure

1. Enter system view.

   `system-view`

2. Specify an IPv6 policy for IPv6 local PBR.

   ipv6 local policy-based-route *policy-name*

   By default, IPv6 local PBR is not enabled.

Specifying an IPv6 policy for IPv6 interface PBR

About this task

Perform this task to apply an IPv6 policy to an interface to guide the forwarding of packets received on the interface only.

Restrictions and guidelines

You can apply only one policy to an interface and must make sure the specified policy already exists. Before you apply a new policy, you must first remove the current policy from the interface.

You can apply a policy to multiple interfaces.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

interface *interface-type interface-number*

3. Specify an IPv6 policy for IPv6 interface PBR.

   ipv6 policy-based-route *policy-name*

   By default, no IPv6 policy is applied to the interface.

Display and maintenance commands for IPv6 PBR

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display IPv6 PBR policy information.	**display ipv6 policy-based-route** [**policy** *policy-name*]
Display IPv6 interface PBR configuration and statistics.	**display ipv6 policy-based-route interface** *interface-type interface-number* [**slot** *slot-number*]
Display IPv6 local PBR configuration and statistics.	**display ipv6 policy-based-route local** [**slot** *slot-number*]
Display IPv6 PBR configuration.	**display ipv6 policy-based-route setup**
Clear IPv6 PBR statistics.	**reset ipv6 policy-based-route statistics** [**policy** *policy-name*]

Contents

Configuring routing policies

About routing policies

Routing policies control routing paths by filtering and modifying routing information.

Routing policies can filter advertised, received, and redistributed routes, and modify attributes for specific routes.

Implementation of a routing policy

To configure a routing policy:

1. Configure filters based on route attributes.
2. Create a routing policy and apply filters to the routing policy.

Filters

Routing policies can use the following filters to match routes.

ACL

An ACL can match the destination or next hop of routes.

For more information about ACLs, see *ACL and QoS Configuration Guide*.

IP prefix list

An IP prefix list matches the destination address of routes.

An IP prefix list can contain multiple items that specify prefix ranges. Each destination IP address prefix of a route is compared with these items in ascending order of their index numbers. A prefix matches the IP prefix list if it matches one item in the list.

AS path list

An AS path list matches the AS_PATH attribute of BGP routes. The AS_PATH attribute identifies the ASs through which a route has passed.

For more information about AS path lists, see "Configuring BGP."

Community list

A community list matches the COMMUNITY attribute of BGP routes. The COMMUNITY attribute identifies the community of BGP routes.

For more information about community lists, see "Configuring BGP."

Extended community list

An extended community list matches the extended community attribute (Route-Target for VPN and Site of Origin) of BGP routes.

RD list

A route distinguisher (RD) list matches the RD of routes.

An RD list is identified by an RD list number and can contain multiple items that specify RD ranges. Each item is identified by an index number. The RD of a route is compared with these items in ascending order of their index numbers. An RD matches the RD list if it matches one item in the list.

Tag list

A tag list matches the tag of IGP routes.

Routing policy

A routing policy can contain multiple nodes, which are in a logical OR relationship. A node with a smaller number is matched first. A route matches the routing policy if it matches one node (except the node configured with the **continue** clause) in the routing policy.

Each node has a match mode of **permit** or **deny**.

- **permit**—Specifies the **permit** match mode for a routing policy node. If a route meets all the **if-match** clauses of the node, it is handled by the **apply** clauses of the node. The route is not compared with the next node unless the **continue** clause is configured. If a route does not meet all the **if-match** clauses of the node, it is compared with the next node.
- **deny**—Specifies the **deny** match mode for a routing policy node. The **apply** and **continue** clauses of a deny node are never executed. If a route meets all the **if-match** clauses of the node, it is denied without being compared with the next node. If a route does not meet all the **if-match** clauses of the node, it is compared with the next node.

A node can contain a set of **if-match**, **apply**, and **continue** clauses.

- **if-match** clauses—Specify the match criteria that match the attributes of routes. The **if-match** clauses of different types are in a logical AND relationship and the **if-match** clauses of the same type are in a logical OR relationship. A route must meet **if-match** clauses of all types to match the node.
- **apply** clauses—Specify the actions to be taken on permitted routes, such as modifying a route attribute.
- **continue** clause—Specifies the next node. A route that matches the current node (permit node) must match the specified next node in the same routing policy. The **continue** clause combines the **if-match** and **apply** clauses of the two nodes to improve flexibility of the routing policy. After you configure a **continue** clause, a route can pass the routing policy even if it does not match the specified next node. To reject such a route, add a **deny** node without clauses.

Follow these guidelines when you configure **if-match**, **apply**, and **continue** clauses:

- If you only want to filter routes, do not configure **apply** clauses.
- If you do not configure any **if-match** clauses for a permit node, the node will permit all routes.
- Configure a permit node containing no **if-match** or **apply** clauses following multiple deny nodes to allow unmatched routes to pass.

Routing policy tasks at a glance

To configure a routing policy, perform the following tasks:

1. (Optional.) Configure filters:
   - Configuring an IPv4 prefix list
   - Configuring an IPv6 prefix list
   - Configuring an AS path list
   - Configuring a community list
   - Configuring an extended community list
   - Configuring an RD list
   - Configuring a tag list
2. Configuring a routing policy:
   a. Creating a routing policy
   b. Configuring if-match clauses

c. Configuring apply clauses
d. Configuring the continue clause

Configuring an IPv4 prefix list

Restrictions and guidelines

If all the items are set to **deny** mode, no routes can pass the IPv4 prefix list. To permit unmatched IPv4 routes, you must configure the **permit** 0.0.0.0 0 **less-equal** 32 item following multiple **deny** items.

Procedure

1. Enter system view.
   `system-view`
2. Configure an IPv4 prefix list.
   `ip prefix-list` *`prefix-list-name`* [**`index`** *`index-number`*] { **`deny`** | **`permit`** } *`ip-address mask-length`* [**`greater-equal`** *`min-mask-length`*] [**`less-equal`** *`max-mask-length`*]

Configuring an IPv6 prefix list

Restrictions and guidelines

If all items are set to **deny** mode, no routes can pass the IPv6 prefix list. To permit unmatched IPv6 routes, you must configure the **permit** :: 0 **less-equal** 128 item following multiple **deny** items.

Procedure

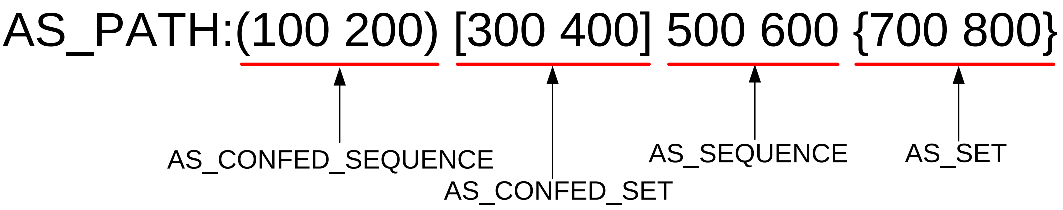
1. Enter system view.
   `system-view`
2. Configure an IPv6 prefix list.
   `ipv6 prefix-list` *`prefix-list-name`* [**`index`** *`index-number`*] { **`deny`** | **`permit`** } *`ipv6-address`* { **`inverse`** *`inverse-prefix-length`* | *`prefix-length`* [**`greater-equal`** *`min-prefix-length`*] [**`less-equal`** *`max-prefix-length`*] }

Configuring an AS path list

About this task

The AS_PATH attribute identifies the ASs through which a BGP route has passed. Figure 1 shows the AS_PATH attribute format in the BGP routing table.

Figure 1 AS_PATH attribute



The AS_PATH attribute value is a character string containing digits, parentheses, brackets, braces, and spaces. It is a sequence of the following AS path segments:

- **AS_CONFED_SEQUENCE**—Ordered set of member AS numbers in the local confederation that the UPDATE message has traversed.
- **AS_CONFED_SET**—Unordered set of member AS numbers in the local confederation that the UPDATE message has traversed.
- **AS_SEQUENCE**—Ordered set of ASs a route in the UPDATE message has traversed.
- **AS_SET**—Unordered set of ASs a route in the UPDATE message has traversed.

NOTE:

The AS_PATH attribute of a BGP route might not contain all of the four AS path segments.

An AS path list consists of regular expressions used for filtering BGP routes by their AS_PATH attributes. A regular expression can contain the special characters described in Table 1.

Table 1 Special characters supported in a regular expression

Character	Description	Example
^	Matches the beginning of a line.	Match a local route if its AS_PATH attribute is null: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } ^$
$	Matches the end of a line.	Match a route if it is originated from AS 100: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 100$
.	Matches any single character.	Match a route if it has passed AS 100 but is not originated or received from AS 100: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } **._100_.**
*	Matches the preceding character or string zero, one, or multiple times.	Match a route if it has an AS_PATH attribute: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } **.***
+	Matches the preceding character or string one or multiple times.	Match a route if its AS_PATH attribute value contains **5**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 5+
x|y	Matches the preceding or succeeding string.	Match a route if it is originated from AS 100 or AS 200: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 100$\|200$
( )	Matches the string in the parentheses, usually used together with the plus sign (+) or asterisk sign (*).	Match a route if its AS_PATH attribute value contains **123**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } (123)+
[xyz]	Matches a single character in the brackets.	Match a route if it is received from AS 10 through AS 19: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } ^1[0-9]$

Character	Description	Example
[^xyz]	Matches a single character that is not in the brackets.	Match a route if its originating AS number does not end with **2** or **4**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } [^24]$
{n}	Matches the preceding character *n* times. The number *n* must be a nonnegative integer.	Match a route if its AS_PATH attribute value contains two **5**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 5{2}
{n,}	Matches the preceding character *n* times or more. The number *n* must be a nonnegative integer.	Match a route if its AS_PATH attribute value contains two or more consecutive **5**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 5{2,}
{n,m}	Matches the preceding character *n* to *m* times or more. The numbers *n* and *m* must be nonnegative integers and *n* cannot be greater than *m*.	Match a route if its AS_PATH attribute value contains one or more consecutive **5**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 5{1,2}
[a-z]	Matches a single character within the specified range for only once.	Match a route if its AS_PATH attribute value contains **0**, **1**, or **2**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** }[0-2]
[^a-z]	Matches a single character that is not in the specified range.	Match a route if its AS_PATH attribute value contains digits other than **0**, **1**, or **2**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** }^ [^0-2]$
_	Matches a punctuation. The expression can begin or end with _.	Match a route if it has passed AS101 100: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } 101_100_
\b	Matches a word that starts with the pattern following \b or ends with the pattern preceding \b.	Match a route if it is originated from AS 100: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } \b100$
\B	Matches a word that contains the pattern but does not start or end with the pattern.	Match a route if its AS_PATH attribute value starts with **1**: **ip as-path** { *as-path-number* | *as-path-name* } { **deny** | **permit** } ^1\B
\	Escape character. If a special character listed in this table follows \, the specific meaning of the character is removed.	N/A
\w	Same as [A-Za-z0-9_], matches a digit, letter, or underscore.	Match a route if its AS_PATH attribute value contains double digits: **ip as-path** { *as-path-number* | *as-path-name* } \w{2}

Character	Description	Example
\W	Same as [^A-Za-z0-9_], matches a character that is not a digit, letter, or underscore.	N/A
\index	Matches the specified string in the parentheses followed by \ twice. Each string in the parentheses is numbered from 1 in order. The *index* argument represents the sequence number of the string you want to match. If the parentheses contain *n* strings, you can specify the sequence number of the string in the range of 1 to *n*.	Match a route if its AS_PATH attribute value contains two consecutive **1**: **ip as-path** { *as-path-number* | *as-path-name* } (1)\1

NOTE:

The regular expressions shown in Table 1 are used for illustration only. Other regular expressions might be available to achieve the same effect.

Restrictions and guidelines

You can configure multiple items for an AS path list that is identified by a number. The relationship between the items is logical OR. A route matches the AS path list if it matches one item in the list.

Procedure

1. Enter system view.
   system-view
2. Configure an AS path list.
   ip as-path *as-path-number* { **deny** | **permit** } *regular-expression*

Configuring a community list

About this task

You can configure multiple items for a community list that is identified by a number. The relationship between the items is logical OR. A route matches the community list if it matches one item in the list.

An advanced community list matches the community attributes of BGP routes based on the regular expressions described in "Configuring an AS path list."

Procedure

1. Enter system view.
   system-view
2. Configure a community list.
   - Configure a basic community list.
     ip community-list { *basic-comm-list-num* | **basic** *basic-comm-list-name* } { **deny** | **permit** } [*community-number*&<1-32> | *aa*:*nn*&<1-32>] [**internet** | **no-advertise** | **no-export** | **no-export-subconfed**] *
   - Configure an advanced community list.
     ip community-list { *adv-comm-list-num* | **advanced** *adv-comm-list-name* } { **deny** | **permit** } *regular-expression*

Configuring an extended community list

About this task

You can configure multiple items for an extended community list that is identified by a number. The relationship between the items is logical OR. A route matches the extended community list if it matches one item in the list.

Procedure

1. Enter system view.

   system-view

2. Configure an extended community list.

   ip extcommunity-list *ext-comm-list-number* [**index** *index-number*] { **deny** | **permit** } { **rt** *route-target* | **soo** *site-of-origin* }&<1-32>

Configuring an RD list

About this task

You can configure multiple items for an RD list that is identified by a number. The relationship between the items is logical OR. A route matches the RD list if it matches one item in the list.

Procedure

1. Enter system view.

   system-view

2. Configure an RD list.

   ip rd-list *rd-list-number* [**index** *index-number*] { **deny** | **permit** } **route-distinguisher**&<1-10>

Configuring a tag list

About this task

You can configure multiple items for a tag list. The relationship between the items is logical OR. A route matches the tag list if it matches one item in the list.

Procedure

1. Enter system view.

   system-view

2. Configure a tag list.

   route tag-list *tag-list-number* [**index** *index-number*] { **deny** | **permit** } *tag-value*&<1-32>

Configuring a routing policy

Creating a routing policy

About this task

A routing policy must have a minimum of one permit node. If all the nodes are in **deny** mode, no routes can pass the routing policy.

Procedure

1. Enter system view.

   system-view

2. Create a routing policy and a node, and enter routing policy node view.

   route-policy *route-policy-name* { **deny** | **permit** } **node** *node-number*

3. (Optional.) Configure a description for the routing policy node.

   description *text*

Configuring if-match clauses

About this task

You can either specify no **if-match** clauses or multiple **if-match** clauses for a routing policy node. If no **if-match** clause is specified for a permit node, all routes can pass the node. If no **if-match** clause is specified for a deny node, no routes can pass the node.

Restrictions and guidelines

When you configure **if-match** clauses, follow these restrictions and guidelines:

- The **if-match** clauses of a routing policy node have a logical AND relationship. A route must meet all **if-match** clauses before it can be executed by the **apply** clauses of the node. If an **if-match** command exceeds the maximum length, multiple **if-match** clauses of the same type are generated. These clauses have a logical OR relationship. A route only needs to meet one of them.
- All IPv4 routes match a node if the **if-match** clauses of the node use only IPv6 ACLs. All IPv6 routes match a node if the **if-match** clauses of the node use only IPv4 ACLs.
- If the ACL used by an **if-match** clause does not exist, the clause is always matched. If no rules of the specified ACL are matched or the match rules are inactive, the clause is not matched.
- If the prefix list, community list, or extended community list used by an **if-match** clause does not exist, the clause is always matched. If no rules of the specified prefix list, community list, or extended community list are matched, the clause is not matched.

Procedure

1. Enter system view.

   system-view

2. Enter routing policy node view.

   route-policy *route-policy-name* { **deny** | **permit** } **node** *node-number*

3. Match routes whose destination, next hop, or source address matches an ACL or prefix list.

   IPv4:

   if-match ip { **address** | **next-hop** | **route-source** } { **acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } | **prefix-list** *prefix-list-name* }

   IPv6:

   if-match ipv6 { **address** | **next-hop** | **route-source** } { **acl** { *ipv6-acl-number* | **name** *ipv6-acl-name* } | **prefix-list** *prefix-list-name* }

   By default, no ACL or prefix list match criterion is configured.

   The ACL specified in an **if-match** clause must be a non-VPN ACL.

4. Configure BGP route match criteria.

   ○ Match BGP routes whose AS_PATH attribute matches a specified AS path list.

   if-match as-path *as-path-number*&<1-32>

- Match BGP routes whose COMMUNITY attribute matches a specified community list.

  if-match community { { *basic-community-list-number* | **name** *comm-list-name* } [**whole-match**] | *adv-community-list-number* }&<1-32>

- Match BGP routes whose extended community attribute matches a specified extended community list.

  if-match extcommunity *ext-comm-list-number*&<1-32>

- Match BGP routes having the specified local preference.

  if-match local-preference *preference*

By default, no BGP route match criteria are configured.

5. Configure route match criteria.

   - Match routes having the specified cost.

     if-match cost *cost-value*

   - Match routes having the specified output interface.

     if-match interface { *interface-type interface-number* }&<1-16>

     This command is not supported by BGP.

   - Match routes having the specified route type.

     if-match route-type { **bgp-evpn-imet** | **bgp-evpn-ip-prefix** | **bgp-evpn-mac-ip** | **external-type1** | **external-type1or2** | **external-type2** | **internal** | **is-is-level-1** | **is-is-level-2** | **nssa-external-type1** | **nssa-external-type1or2** | **nssa-external-type2** } *

   - Match IGP routes having the specified tag value.

     if-match tag *tag-value*

   - Match IGP routes whose tag matches the specified tag list.

     if-match tag-list *tag-list-number*

   By default, no route match criteria are configured.

6. Match routes whose RD matches the specified RD list.

   if-match rd-list *rd-list-number*

   By default, no RD match criterion is configured.

Configuring apply clauses

1. Enter system view.

   system-view

2. Enter routing policy node view.

   route-policy *route-policy-name* { **deny** | **permit** } **node** *node-number*

3. Configure BGP route attributes.

   - Set the AS_PATH attribute for BGP routes.

     apply as-path *as-number*&<1-32> [**replace**]

   - Delete the specified COMMUNITY attribute for BGP routes.

     apply comm-list { *comm-list-number* | *comm-list-name* } **delete**

     By default, no COMMUNITY attribute is deleted for BGP routes.

   - Set the specified COMMUNITY attribute for BGP routes.

apply community { **none** | **additive** | { *community-number*&<1-32> | *aa:nn*&<1-32> | **internet** | **no-advertise** | **no-export** | **no-export-subconfed** } * [**additive**] }

- Set the extended community attribute for BGP routes.

  apply extcommunity { **rt** *route-target* }&<1-32> [**additive**]
- Set a local preference for BGP routes.

  apply local-preference *preference*
- Set the ORIGIN attribute for BGP routes.

  apply origin { **egp** *as-number* | **igp** | **incomplete** }
- Set a preferred value for BGP routes.

  apply preferred-value *preferred-value*
- Set a traffic index for BGP routes.

  apply traffic-index { *value* | **clear** }

By default, no BGP route attributes are configured.

4. Configure the route cost and cost type.
   - Set a cost for routes.

     apply cost [**+** | **-**] *cost-value*

     By default, no cost is set for routes.
   - Set a cost type for routes.

     apply cost-type { **external** | **internal** | **type-1** | **type-2** }

     By default, no cost type is set for routes.
5. Set the next hop for routes.

   IPv4:

   apply ip-address next-hop *ip-address* [**public** | **vpn-instance** *vpn-instance-name*]

   IPv6:

   apply ipv6 next-hop *ipv6-address*

   By default, no next hop is set for routes.

   The configuration does not apply to redistributed routes.
6. Configure route priorities.
   - Set an IP precedence for matching routes.

     apply ip-precedence { *value* | **clear** }

     By default, no IP precedence is set.
   - Set a preference.

     apply preference *preference*

     By default, no preference is set.
   - Set a prefix priority.

     apply prefix-priority { **critical** | **high** | **medium** }

     By default, the prefix priority is low.
7. Redistribute routes to the specified IS-IS level.

   apply isis { **level-1** | **level-1-2** | **level-2** }

   By default, routes are not redistributed into the specified IS-IS level.
8. Set a local QoS ID for matching routes.

   apply qos-local-id { *local-id-value* | **clear** }

By default, no local QoS ID is set.

9. Set a tag value for IGP routes.

   apply tag *tag-value*

   By default, no tag value is set for IGP routes.

10. Set a backup link for fast reroute (FRR).

    IPv4:

    apply fast-reroute { **backup-interface** *interface-type interface-number* [**backup-nexthop** *ip-address*] | **backup-nexthop** *ip-address* }

    IPv6:

    apply ipv6 fast-reroute { **backup-interface** *interface-type interface-number* [**backup-nexthop** *ipv6-address*] | **backup-nexthop** *ipv6-address* }

    By default, no backup link is set for FRR.

Configuring the continue clause

Restrictions and guidelines

When you configure the **continue** clause to combine multiple nodes, follow these restrictions and guidelines:

- If you configure an **apply** clause that sets different attribute values on all the nodes, the **apply** clause of the node configured most recently takes effect.
- If you configure the following **apply** clauses on all the nodes, the **apply** clause of each node takes effect:
  - **apply as-path** without the **replace** keyword.
  - **apply cost** with the **+** or **–** keyword.
  - **apply community** with the **additive** keyword.
  - **apply extcommunity** with the **additive** keyword.
- The **apply comm-list delete** clause configured on the current node cannot delete the community attributes set by the **apply community** clauses of the preceding nodes.

Procedure

1. Enter system view.

   system-view

2. Enter routing policy node view.

   route-policy *route-policy-name* { **deny** | **permit** } **node** *node-number*

3. Specify the next node to be matched.

   continue [*node-number*]

   By default, no continue clause is configured.

   The specified next node must have a larger number than the current node.

Display and maintenance commands for routing policies

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display BGP AS path list information.	**display ip as-path** [*as-path-number*]
Display BGP community list information.	**display ip community-list** [*basic-community-list-number* | *adv-community-list-number* | **name** *comm-list-name*]
Display BGP extended community list information.	**display ip extcommunity-list** [*ext-comm-list-number*]
Display IPv4 prefix list statistics.	**display ip prefix-list** [**name** *prefix-list-name*]
Display RD list information.	**display ip rd-list** [*rd-list-number*]
Display IPv6 prefix list statistics.	**display ipv6 prefix-list** [**name** *prefix-list-name*]
Display tag list information.	**display route tag-list** [*tag-list-number*]
Display routing policy information.	**display route-policy** [**name** *route-policy-name*]
Clear IPv4 prefix list statistics.	**reset ip prefix-list** [*prefix-list-name*]
Clear IPv6 prefix list statistics.	**reset ipv6 prefix-list** [*prefix-list-name*]

Contents

Configuring guard routes

About guard routes

A guard route directs traffic to the guard device for filtering and cleaning. You can manually configure a guard route on the guard device, or use a script to automatically configure a guard route upon receipt of a notification.

Guard route characteristics

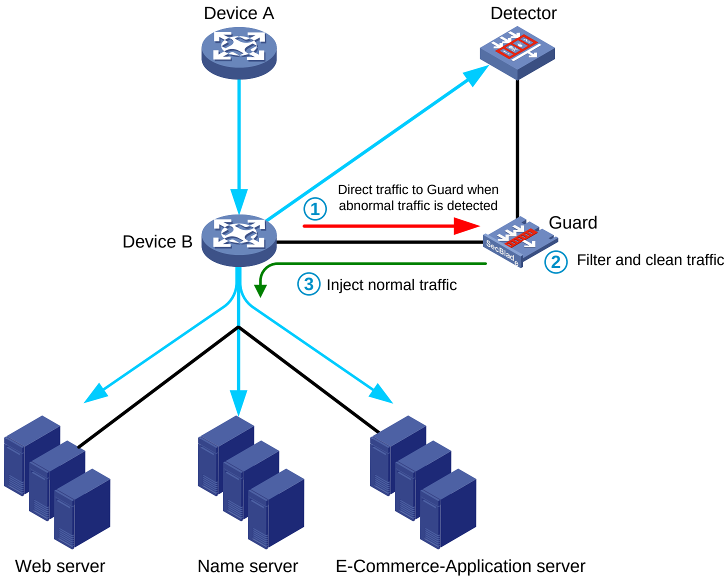
Guard routes use Null 0 as the outgoing interface.

Guard routes are inactive routes and will not be installed into the FIB.

You must configure a routing protocol, such as BGP, OSPF, or OSPFv3, to redistribute and advertise guard routes for directing traffic to the guard device.

Guard route mechanism

Figure 1 Guard route network diagram



As shown in Figure 1, Device B uses port mirroring to mirror traffic destined for the application servers to the detector for monitoring. It uses a routing protocol to redistribute guard routes from the guard device.

When no abnormal traffic is detected, Device B forwards traffic without the participation of the guard device.

When abnormal traffic is detected, the detector performs one of the following tasks:

- Sends a notification to the guard device. The guard device then automatically generates a guard route.
- Generates an alarm to notify the network administrator. The administrator then manually configures a guard route on the guard device.

The destination address of the guard route is the address to which the abnormal traffic is destined.

After the guard route is configured, the guard device advertises the route to Device B. Device B directs all traffic destined to the destination address of the guard route to the guard device. The guard device filters and cleans the traffic and then sends the normal traffic back to Device B.

Restrictions and guidelines: guard route configuration

A guard device is typically used for traffic filtering and cleaning. To avoid system consumption, configure a routing policy on the guard device or its connected device to disable receiving and advertising non-guard routes. For more information about routing policies, see "Configuring routing policies."

Configuring a guard route

1. Enter system view.

   system-view

2. Configure a guard route.

   IPv4:

   ip route-guard *ip-address* { *mask-length* | *mask* }

   By default, no IPv4 guard routes exist.

   IPv6:

   ipv6 route-guard *ipv6-address prefix-length*

   By default, no IPv6 guard routes exist.

Display and maintenance commands for guard routes

Execute **display** commands in any view.

Task	Commands
Display IPv4 guard route information.	**display ip routing-table protocol guard** [**inactive** | **verbose**]
Display IPv6 guard route information.	**display ipv6 routing-table protocol guard** [**inactive** | **verbose**]

Contents

Configuring RIR

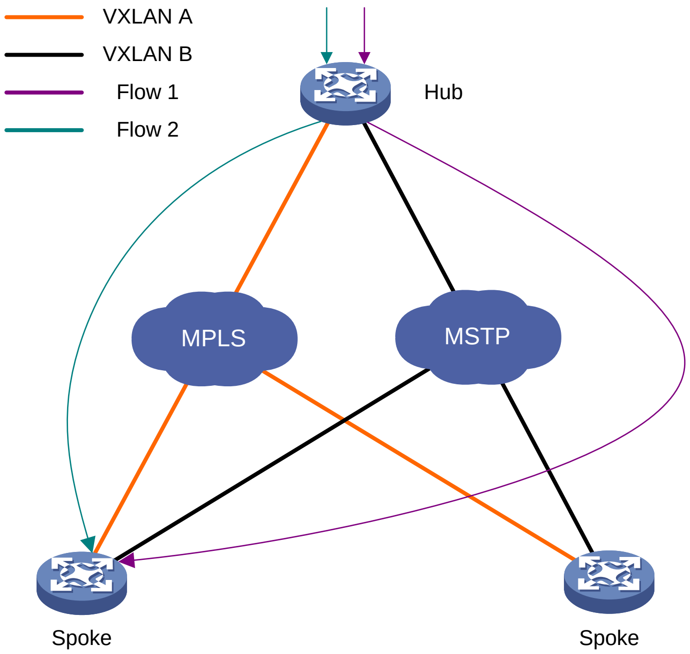
About RIR

Resilient Intelligent Routing (RIR) dynamically selects the most suitable links for traffic forwarding based on service requirements (for example, link quality and link bandwidth). RIR not only can select the optimal link from a specific type of transport network, but also can perform automatic link switchover when the current link becomes unqualified.

Application scenario

As shown in Figure 1, RIR is used in a VXLAN-based hub-spoke network. The feature can select different VXLAN tunnels to forward traffic for different services, depending on parameters including the link preference, link primary/backup role, link quality, and link bandwidth. RIR can perform link selection not only for traffic from a hub to a spoke, but also for traffic from a spoke to a hub. The feature might select different links for bidirectional traffic between a hub and a spoke.

Figure 1 RIR application scenario



Flow template

A flow template defines link selection policies for a type of service flow. A flow ID uniquely identifies a flow template.

The device applies the link selection policies under a flow template to the service flow marked with the flow ID of the flow template.

The device supports using QoS policies to mark flow IDs for service flows. After QoS identifies the service of a packet based on the quintuple and DSCP of the packet, it assigns a flow ID to the packet. Then, RIR will perform link selection for the packet based on the flow template that uses the flow ID.

The flow ID is marked only in the RIR process, and it will not be added to any outgoing packets.

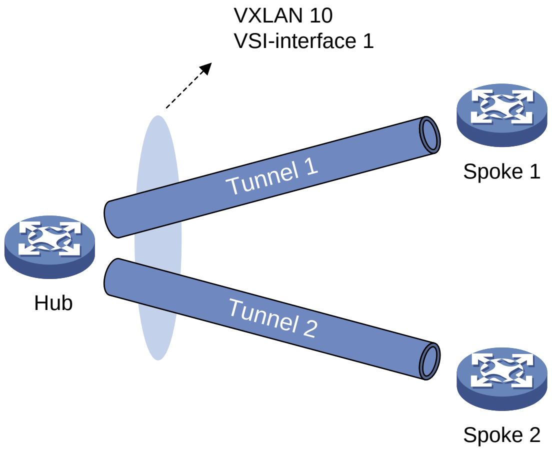
For more information about QoS marking, see QoS overview, QoS policies, and marking configuration in *ACL and QoS Configuration Guide*.

Link type

RIR uses the link type to identify the network type of a link and uses the link index to distinguish links of the same network type. RIR supports 4G, Internet, MPLS, and MSTP link types. The link type is used only for identifying links, and it does not affect packet encapsulation.

As shown in Figure 2, assign a link type and link index to the VSI interface to identify the VXLAN tunnels between the hub and the spokes in the VXLAN. To uniquely identify a VXLAN tunnel between a hub and a spoke, VXLAN-based RIR allows the hub and spoke to have only one VXLAN tunnel for each VSI interface.

Figure 2 Links in a VXLAN network



Preference-based link selection

Link preference

You can assign a preference to a link based on factors such as the service requirements, the link conditions, and the link cost. RIR preferentially selects links with higher preference.

VXLAN-based RIR supports assigning a link preference to a type of links with a specific link index in flow template view. The link type and link index identify a VSI interface. As VXLAN-based RIR allows a hub and spoke to have only one VXLAN tunnel for a VSI interface, the link preference of the VSI interface is the link preference of the VXLAN tunnel between the hub and spoke.

Link selection rules

You can assign the same preference value to different links in the same flow template.

RIR selects a link for a type of service flows from the links in the flow template in descending order of link preference. If the links with the highest preference cannot meet the service requirements, RIR tries the links with the second highest preference, and so forth to the links with the lowest preference.

If the flow template has two or more links with the same preference, RIR performs link selection based on RIR link load sharing criteria.

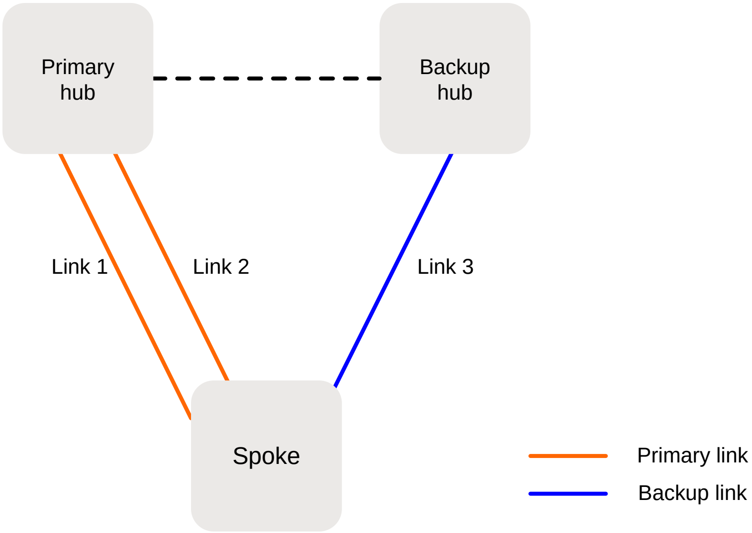
Redundant link selection

To ensure service high availability, redundant hubs are deployed. Typically, use the links between a spoke and the primary hub as primary links, and use the links between a spoke and the backup hub as backup links. When no link is available to reach the primary hub, the spoke can switch traffic to the backup hub to ensure service continuity.

A VXLAN tunnel is a primary link by default after it is assigned a link preference based on the link type and index on its VSI interface. To specify a VXLAN tunnel associated with a VSI interface as a backup link, specify that VXLAN tunnel as an RIR backup tunnel.

As shown in Figure 3, when primary links 1 and 2 are not available, the spoke uses backup link 3 to forward traffic to the backup hub.

Figure 3 Primary and backup links



Quality-based link selection

RIR server and RIR client

About the RIR server and RIR client

In a hub-spoke network, a hub is typically connected to multiple spokes. To avoid the hub from consuming too many resources on link quality detection by Network Quality Analyzer (NQA) probes, RIR provides the following roles:

- **RIR server**—An RIR server does not perform NQA link probes. It performs link selection based on the link quality probe results synchronized from RIR clients.
- **RIR client**—An RIR client performs NQA link probes to detect the link quality and synchronizes the link quality probe results to RIR servers.

Configure a hub as an RIR server and spokes as RIR clients, so the hub can perform link selection based on the link quality probe results synchronized from the RIR clients.

RIR server function

You can enable the RIR server globally or on an interface, as shown in Figure 4.

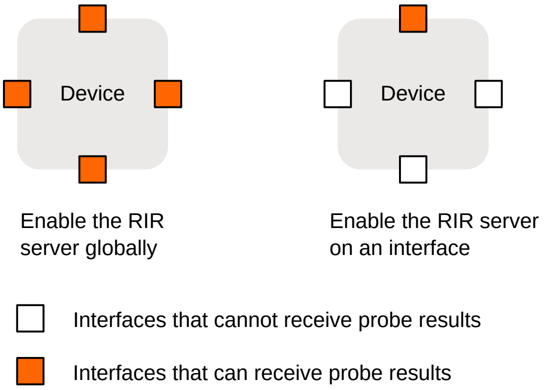
- If you enable the RIR server globally, the RIR server is also enabled on all interfaces on the device. The interfaces can receive RIR link quality probe results from RIR clients.

- If you enable the RIR server on an interface, only that interface can receive RIR link quality probe results from RIR clients.

IMPORTANT:

In a VXLAN network, only tunnel interfaces support enabling the RIR server. The RIR server uses the tunnel interfaces to receive RIR link quality probe results from RIR clients.

Figure 4 Enabling the RIR server


Device
Device
Enable the RIR server globally
Enable the RIR server on an interface
□ Interfaces that cannot receive probe results
■ Interfaces that can receive probe results

RIR client function

The RIR client synchronizes link quality probe results to RIR servers. Enabling the RIR client is the same as enabling the RIR server.

RIR server and client enabling policy

Enable the RIR server or RIR client, or use them in combination, depending on the role of the device in the network.

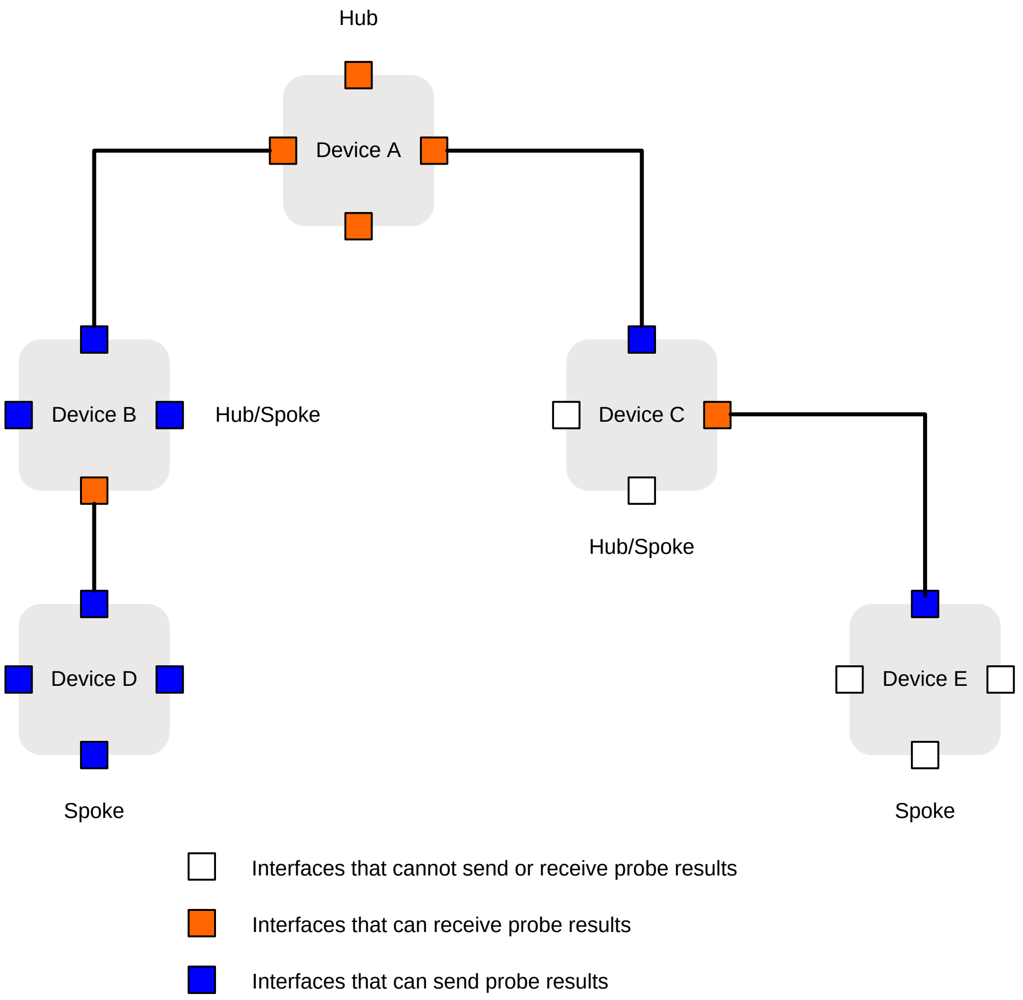
- If the device acts only as a hub, you can enable the RIR server globally.
- If the device acts only as a spoke, you can enable the RIR client globally.
- If the device acts as both a hub and a spoke, you can enable the RIR server and RIR client on the corresponding interfaces.

The RIR server and RIR client cannot be both enabled on the same interface. If the enabled role (RIR server or client) on an interface is different from the globally enabled role, the interface-specific role takes effect on that interface.

As shown in Figure 5, enable the global and interface-specific RIR server and RIR client in combination as follows:

- **Device A**—Enable the RIR server globally.
- **Device B**—Enable the RIR client globally and enable the RIR server on an interface.
- **Device C**—Enable the RIR client on an interface and enable the RIR server on an interface.
- **Device D**—Enable the RIR client globally.
- **Device E**—Enable the RIR client on an interface.

Figure 5 Enabling the RIR server and client


Hub
Device A
Device B
Hub/Spoke
Device C
Hub/Spoke
Device D
Spoke
Device E
Spoke
□ Interfaces that cannot send or receive probe results
🟧 Interfaces that can receive probe results
🟦 Interfaces that can send probe results

NQA link probes

RIR uses NQA to detect the status of candidate links and selects the most suitable link based on the NQA link quality probe results. For more information about NQA, see *Network Management and Monitoring Configuration Guide*.

A hub performs link selection based on the link quality probe results synchronized from spokes. RIR uses spokes as NQA clients and hubs as NQA servers. The following types of NQA link probe operations are defined:

- **NQA link connectivity probe operation**—Performs ICMP echo probes to check the connectivity of each link. If a link is disconnected, RIR does not perform NQA link quality probes on that link. You can configure only one NQA link connectivity probe operation.
- **NQA link quality probe operations**—Also referred to as NQA link quality operations, perform UDP jitter probes to detect the link delay, jitter, and packet loss ratio for links that pass NQA link connectivity check. You can configure multiple NQA link quality operations. The operations might offer different link quality probe results for the same link.

The device performs NQA probes only for links that are assigned link types and link indexes.

SLA

To meet the differentiated requirements of services on link quality, configure a Service Level Agreement (SLA) for each service. An SLA contains a set of link quality evaluation thresholds, including the link delay threshold, packet loss threshold, and jitter threshold.

The quality policy of a flow template contains an SLA and an NQA link quality operation. By comparing the NQA link quality probe results with the thresholds in the SLA, the device determines whether a link meets the quality requirements of the service. If all parameter values in the probe results of a link are lower than or equal to the thresholds in the SLA, the link is qualified for the service.

Quality policy

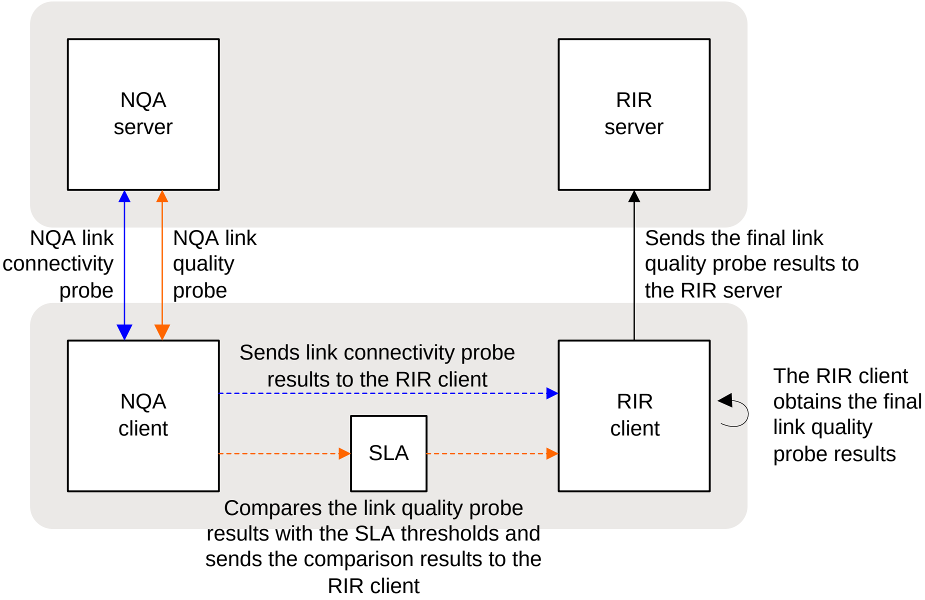
Link quality probe results

As shown in Figure 6, if a flow template is configured with a quality policy, the spoke determines whether a link is qualified for that type of service flow based on the NQA link quality probe results. In addition, the spoke synchronizes the link quality probe results to the hub. The hub performs link selection based on the link quality probe results synchronized from the spoke. You only need to configure the quality policy on the spoke.

RIR uses the following rules to determine whether a link is qualified for a type of service flow:

- If the link fails NQA link connectivity check, RIR determines that the link is unqualified.
- If the link passes NQA link connectivity check, RIR checks the NQA link quality probe results for the link.
    - If all link quality parameter values in the probe results are lower than or equal to the link quality thresholds in the SLA, RIR determines that the link is qualified.
    - If any link quality parameter value in the probe results is higher than the corresponding link quality threshold in the SLA, RIR determines that the link is unqualified.

Figure 6 Link quality probe result processing network diagram


NQA server
RIR server
NQA link connectivity probe
NQA link quality probe
Sends the final link quality probe results to the RIR server
Sends link connectivity probe results to the RIR client
NQA client
SLA
RIR client
The RIR client obtains the final link quality probe results
Compares the link quality probe results with the SLA thresholds and sends the comparison results to the RIR client

Quality-based link selection on a hub and a spoke

When a spoke performs quality-based link selection, it considers only the quality policy configured on the spoke. If a quality policy has been configured in a flow template on the spoke, the spoke calculates link quality probe results based on the NQA link quality operation and SLA in the quality policy. Then, the spoke performs link selection for the flow that matches the flow template based on the link quality probe results. If no quality policy is configured in a flow template on the spoke, the spoke does not consider the quality factor when it performs link selection. All links in the flow template meet the service requirements in quality.

When a hub performs quality-based link selection for traffic sent to a spoke, it considers the quality policies both on the hub and spoke. Table 1 shows the link selection rules for a flow that matches a flow template on a hub.

Table 1 Quality-based link selection rules on a hub

Whether a quality policy is configured on the hub	Whether a quality policy is configured on the spoke	Quality-based link selection rules on the hub
Yes	Yes	The hub performs link selection based on the link quality probe results synchronized from the spoke.
Yes	No	The spoke does not synchronize link quality probe results to the hub. The hub determines that no link in the flow template meets the quality requirements.
No	Yes	The spoke synchronizes link quality probe results to the hub. The hub determines that all links in the flow template are qualified for packets that match the flow template.
No	No	The spoke does not synchronize link quality probe results to the hub. The hub determines that all links in the flow template are qualified for packets that match the flow template.

As a best practice, configure a quality policy both on the hub and spoke for a type of service flow or do not configure any quality policy on the hub or spoke for the type of service flow.

Bandwidth-based link selection

About bandwidth-based link selection

Bandwidth-based link selection not only can select links that meet the service bandwidth requirements, but also can load share service traffic among multiple links. This manner can avoid a link from being overwhelmed or congested.

Bandwidth-based link selection policy

Bandwidth-based link selection enables the device to select a suitable link for service traffic based on the used bandwidth of the link, the total link bandwidth, and the per-session expected bandwidth. The device can automatically obtain the used bandwidth of the link. The total bandwidth of the link and the per-session expected bandwidth are manually configured or calculated based on the user configuration.

The device uses sessions as the minimum granularity and performs bandwidth-based link selection to achieve refined link bandwidth management. A session is uniquely defined by a quintuple including the source IP address, destination IP address, source port, destination port, and transport layer protocol.

When the device selects links for traffic of a session, it first performs bandwidth detection based on the per-session expected bandwidth in the flow template to which the session belongs. If the used bandwidth plus the per-session expected bandwidth of a candidate link is less than 80% of its total bandwidth, the available bandwidth of the link meets the session bandwidth requirements. The link passes the bandwidth detection.

When different sessions of the same service type use the same link selection policy, the link selection results might be different.

Load balancing

Load balancing modes

If multiple links are available for sessions that match a flow template, the device distributes the traffic of the sessions to these links for load balancing based on the link bandwidths. RIR supports the following load balancing modes:

- **Per-session weight-based link selection mode**—RIR global link load balancing mode that takes effect on all RIR flows. This mode can distribute the sessions that match the same flow template to different links according to the weights of the links. RIR selects only one link to transmit a session.
- **Per-session periodic link adjustment mode**—RIR global link load balancing mode that takes effect on all RIR flows. This mode not only can distribute the sessions that match the same flow template to different links, but also can periodically adjust links for the sessions. Within one adjustment period, RIR selects only one link to transmit a session.
- **Per-packet mode**—Flow-specific link load balancing mode that takes effect only on traffic that matches the flow template where this mode is enabled. This mode can distribute the same session to different links for transmission.

The per-packet mode takes precedence over the per-session modes.

Load balancing concepts

Bandwidth weight

The bandwidth of a link is used as the weight of the link. If the total bandwidth of a link is used as the link weight, the weight is called the total bandwidth weight. If the remaining bandwidth of a link is used as the link weight, the weight is called the remaining bandwidth weight.

To select links from multiple links based on their bandwidth weights, the probability that each link is selected equals to the ratio of the bandwidth of a single link to the bandwidth sum of all the links. The bandwidth sum of all the links is the weight sum.

For example, the total bandwidths of links 1, 2, and 3 are 10 Mbps, 10 Mbps, and 20 Mbps, respectively. The remaining bandwidths of links 1, 2, and 3 are 8 Mbps, 4 Mbps, and 8 Mbps, respectively. All the links meet the service requirements. To select links based on the total bandwidth weights, the probabilities that links 1, 2, and 3 are selected are 25%, 25%, and 50%, respectively. To select links based on the remaining bandwidth weights, the probabilities that links 1, 2, and 3 are selected are 40%, 20%, and 40% ,respectively.

Remaining bandwidth ratio

The remaining bandwidth ratio refers to the ratio of the remaining bandwidth of a link compared to the total bandwidth of the link. If the remaining bandwidth ratio of a link is the largest among multiple links, this remaining bandwidth ratio is called the largest remaining bandwidth ratio. If the remaining bandwidth ratio of a link is the smallest among multiple links, this remaining bandwidth ratio is called the smallest remaining bandwidth ratio.

Per-session weight-based link selection mode

The mechanisms of the per-session weight-based link selection mode are as follows:

- **For preference-based primary link selection, preference-based backup link selection, and quality tolerant link selection**—If multiple links with the same preference meet the requirements of a flow template, RIR selects one optimal link for each session of the flow template from these links. In this mode, RIR performs link selection based on the remaining bandwidth weight of each link. The used bandwidth is the actually used bandwidth plus the per-session expected bandwidth.
- **For bandwidth tolerant link selection**—If multiple links meet the requirements of a flow template, RIR selects one optimal link for each session of the flow template from these links. In this scenario, RIR performs link selection based on the total bandwidth weight of each link.

For more information about the preference-based primary or backup link selection, quality tolerant link selection, and bandwidth tolerant link selection, see "RIR working mechanisms."

Per-session periodic link adjustment mode

The mechanisms of the per-session periodic link adjustment mode are as follows:

- **For preference-based primary link selection, preference-based backup link selection, and quality tolerant link selection**—If multiple links with the same preference meet the requirements of a flow template, RIR selects one optimal link for each session of the flow template from these links. RIR preferentially selects the link with the lowest bandwidth usage for a session. The used bandwidth is the actually used bandwidth plus the per-session expected bandwidth.
- **For bandwidth tolerant link selection**—If multiple links meet the requirements of a flow template, RIR selects one optimal link for each session of the flow template from these links. The link selected the last time for a session takes precedence over the other links for that session. If RIR performs link selection for a session for the first time, it selects a link based on the remaining bandwidth weights of the available links.

In per-session periodic link adjustment mode, the device periodically detects the bandwidth usage of all links that have RIR sessions at the configured adjustment intervals. RIR reselect links for sessions that match a flow template if the links in the flow template meet the following requirements: The difference between the largest remaining bandwidth ratio and the smallest remaining bandwidth ratio becomes larger than or equal to the periodic adjustment upper threshold. The link adjustment might be last for several adjustment intervals. RIR stops link adjustment if one of the following requirements is met:

- The difference between the largest remaining bandwidth ratio and the smallest remaining bandwidth ratio of the links becomes smaller than the periodic adjustment lower threshold.
- The adjustment interval is the 20th interval after link reselection is triggered.

Per-packet mode

The mechanisms of the per-packet mode are as follows:

- **For preference-based primary link selection, preference-based backup link selection, and quality tolerant link selection**—If multiple links with the same preference meet the requirements of a session, all these links are candidate optimal links for this session. When forwarding traffic for the session, the device distributes the traffic to these links packet by packet according to the remaining bandwidth weight of each link.

  For example, the device needs 10 Mbps of bandwidth to transmit traffic for a session with flow ID 1. Links 1 and 2 are available. The remaining bandwidth of link 1 is 20 Mbps and the remaining bandwidth of link 2 is 30 Mbps. Finally, the traffic of this session uses 4 Mbps of bandwidth on link 1 and 6 Mbps of bandwidth on link 2.

- **For bandwidth tolerant link selection**—If multiple links meet the requirements of a session, all these links are candidate optimal links for this session. When forwarding traffic for the session, the device distributes the traffic to these links packet by packet. Each link has the same probability to be selected.

Link selection delay and suppression

To improve packet forwarding efficiency, the device does not repeatedly perform link selection for traffic of the same session. After the device performs link selection for traffic of a session, it forwards the subsequent traffic of that session according to the previous link selection result. Link reselection is triggered when any link in the session's flow template has one of the following changes:

- The quality of a link becomes qualified from unqualified or the quality of a link becomes unqualified from qualified.
- The bandwidth usage of a link has reached the maximum.

To avoid frequent link selection caused by link flapping, RIR defines a link selection delay and link selection suppression period.

After the device performs link selection, it starts the link selection suppression period if the period has been configured. Within the link selection suppression period, the device does not perform link reselection, but it maintains the link state data. When the link selection suppression period ends, the link selection delay timer starts. If the link state still meets the conditions that can trigger link reselection when the delay timer expires, the device performs link reselection. If the link state changes to not meet the conditions that can trigger link reselection within the delay time, the device does not perform link reselection.

RIR collaboration

About RIR collaboration

RIR collaboration enables multiple RIR-capable devices to share link data, load share traffic, and realize distributed link schedule through establishing RIR dedicated links between each pair of them.

Network structure

As shown in Figure 7, RIR collaboration has the following concepts:

- **RIR collaboration device group**—Contains a group of devices that collaboratively select links. Each pair of devices in an RIR collaboration device group are peer devices to each other. They share links for RIR.

  Typically, devices in the same RIR collaboration device group are deployed in the same physical area, for example, the same equipment room or campus.

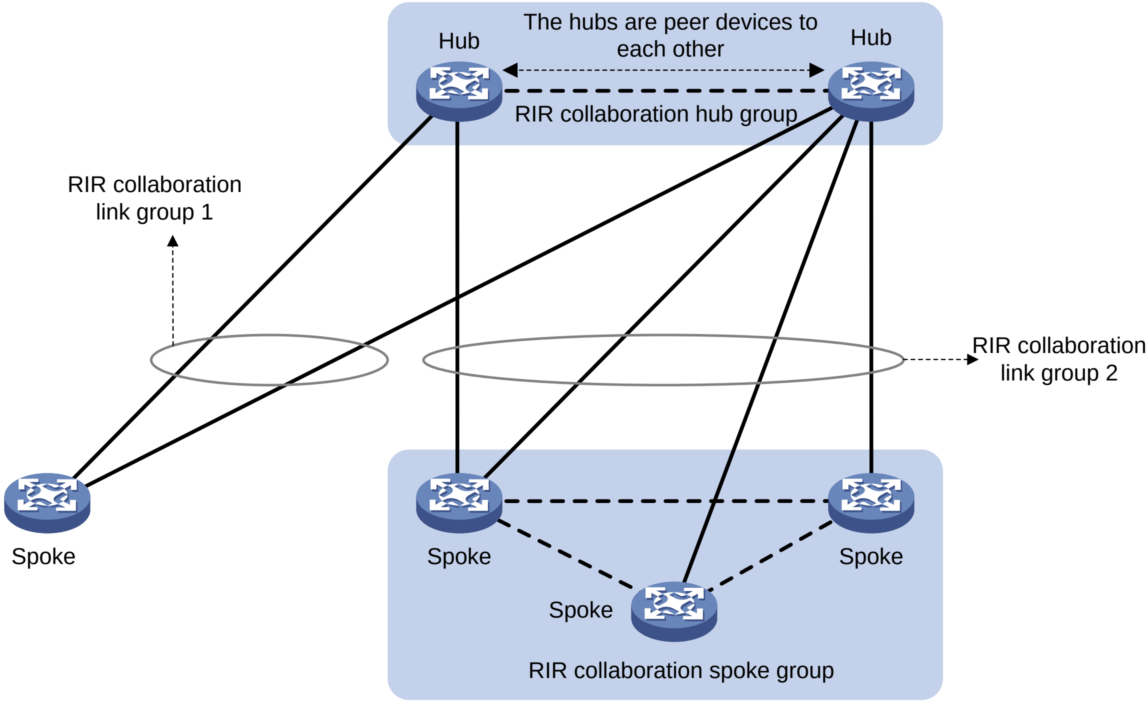
  - A group of hubs can form an RIR collaboration hub group.
  - A group of spokes can form an RIR collaboration spoke group.

  An RIR collaboration device group acts as a logical device. The logical device can form a hub-spoke network with RIR collaboration device groups or physical devices.

- **RIR collaboration link group**—Contains a group of links that connect an RIR collaboration device group and another RIR collaboration device group or a physical device (hub or spoke). As shown in Figure 7, RIR collaboration link groups 1 and 2 connect the RIR collaboration hub group to a spoke and an RIR collaboration spoke group, respectively.
- **Local packets and peer packets**—In RIR collaboration, the packets firstly processed by the local device are called local packets and the packets firstly processed by the peer device are called peer packets. Devices in an RIR collaboration device group use different link selection

policies to process local packets and peer packets. If a device does not belong to an RIR collaboration device group, all service packets received by the device are local packets.

Figure 7 RIR collaboration network structure


Hub
The hubs are peer devices to each other
Hub
RIR collaboration hub group
RIR collaboration link group 1
RIR collaboration link group 2
Spoke
Spoke
Spoke
Spoke
RIR collaboration spoke group

Working mechanism

The RIR collaboration mechanism is as follows:

1. After each pair of devices in an RIR collaboration device group establish RIR collaboration relationship, the device with a lower IP address acts as the client. The client initiates a TCP connection to the peer device.
2. Through TCP connections, a device synchronizes the configuration and status data of local links that meet the service requirements to its peer devices. The data does not include the link data synchronized from other devices in the same RIR collaboration device group. As a result, the devices in the RIR collaboration device group can obtain link information from one another, and can update link information in real time.
3. When a device in the RIR collaboration device group receives a service packet for a session, it handles the packets as shown in Table 2.

Table 2 Collaborative link selection policies

Packet condition	Are routes available to forward the packet	Collaborative link selection policies
Local packet and received for the first time	Yes	If the optimal link information exists for the session, the local device forwards the packet according to the information. If no optimal link information exists for the session, the local device examines all links in the same flow template on the local device and peer devices. Then, the local device selects the most suitable link for the packet. • If the selected link is a link on the local device, the local device directly forwards the packet.

Packet condition	Are routes available to forward the packet	Collaborative link selection policies
		• If the selected link is a link on a peer device, the local device forwards the packet to the peer device.
Local packet and received for the first time	No	The local device selects a peer device in the same RIR collaboration device group and forwards the packet to the peer device.
Local packet but not received for the first time	Yes	The packet is forwarded to a peer device, and then the peer device returns the packet back to the local device because it does not have a route to forward the packet. After the packet is returned back, the local device only examines the local links in order to find the most suitable link to forward the packet. The local device performs link reselection for the session every 60 seconds. This ensures that the traffic of the session can be switched to the peer device in time after the routes on the peer device recover.
Local packet but not received for the first time	No	The local device selects a peer device to which it has never forwarded the traffic of the session and forwards the packet to the peer device. If no such a peer device is available, the local device discards the packet.
Peer packet	Yes	If the optimal link information exists for the session, the local device forwards the packet according to the information. If no optimal link information exists for the session, the local device selects the most suitable link from the local links to forward the packet.
Peer packet	No	The local device returns the packet back to the original peer device.

RIR working mechanisms

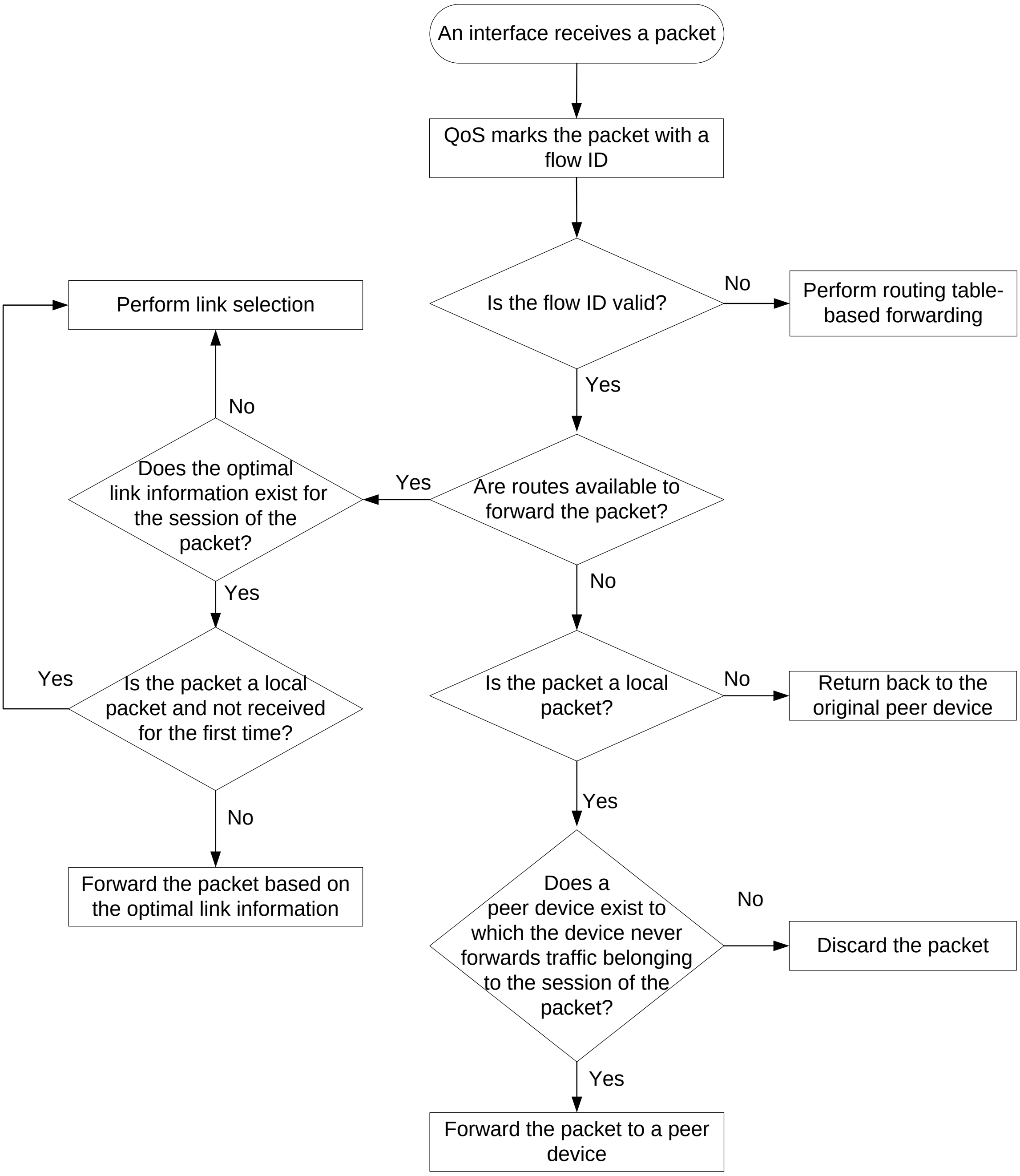
Preparation

When the device receives a packet on an interface, it handles the packet as follows:

1. Uses QoS to mark the packet with a flow ID based on the quintuple and DSCP of the packet.
2. Identifies whether the flow ID of the packet is valid.
   - If the flow ID is invalid, the device performs routing table-based forwarding for the packet.
   - If the flow ID is valid, go to the next step.
3. Performs a routing table lookup to identify whether routes are available to forward the packet.
   - If no route is available, the device identifies whether the packet is a local packet.
     - If the packet is not a local packet, the device returns the packet back to the original peer device.
     - If the packet is a local packet, the device selects a peer device to which it has never forwarded traffic for the packet's session and forwards the packet to the peer device. If no such a peer device is available, the device discards the packet.
   - If routes are available, go to the next step.
4. Examines whether the optimal link information exists for the session of the packet.

- If the optimal link information exists, the device forwards the packet based on the information.
- If no optimal link information exists, the device performs link selection for the packet.

Figure 8 RIR preparation


An interface receives a packet
QoS marks the packet with a flow ID
Is the flow ID valid?
No
Perform routing table-based forwarding
Yes
Are routes available to forward the packet?
Yes
Does the optimal link information exist for the session of the packet?
No
Perform link selection
Yes
Is the packet a local packet and not received for the first time?
Yes
No
Forward the packet based on the optimal link information
No
Is the packet a local packet?
No
Return back to the original peer device
Yes
Does a peer device exist to which the device never forwards traffic belonging to the session of the packet?
No
Discard the packet
Yes
Forward the packet to a peer device

Link selection workflow summary

RIR uses the following workflow to select a link to forward a packet:

1. Selects the flow template that has the same flow ID as the packet.
2. Selects the most suitable link from the links in the flow template by using the following criteria in order:
   a. Preference-based primary link selection.

   b. Preference-based backup link selection.
   c. Quality tolerant link selection.
   d. Bandwidth tolerant link selection.

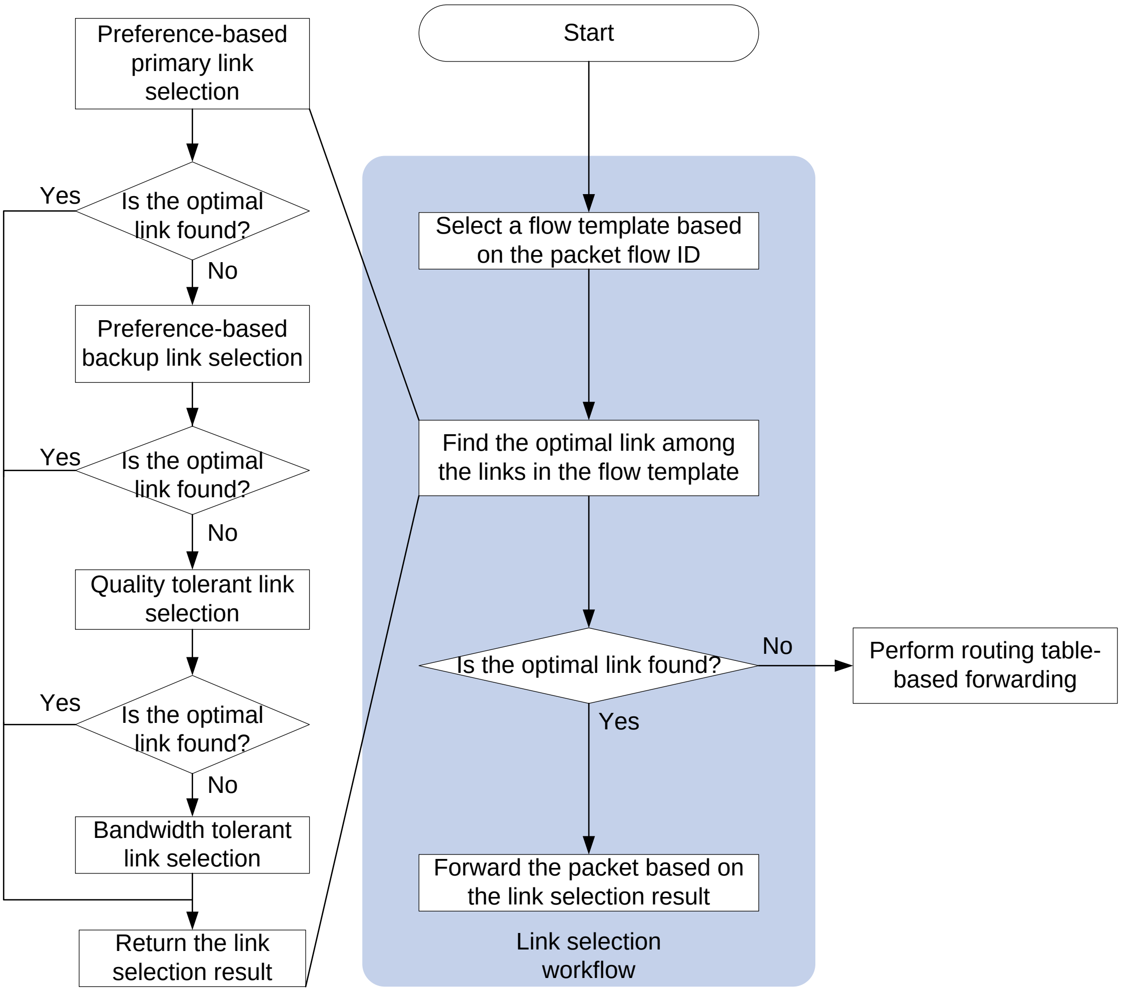
   If the packet is a local packet and is received for the first time, the candidate links also include the links synchronized from RIR collaboration peer devices in the same flow template.

   If the packet is a peer packet or the packet is a local packet but is not received for the first time, the candidate links only include links on the local device.

3. If a link is found suitable, RIR returns the link selection result and stops searching other links. If no link is found suitable for a criterion, RIR uses the next criterion to select links. If RIR fails to find a suitable link by using all criteria, it determines that no link is suitable and returns the link selection result.
4. If no suitable link is found, the device performs forwarding based on the routing table. If a suitable link is found, RIR forwards the packet based on the link selection result.
   - If the most suitable link belongs to the local device, the device directly forwards the packet through the link.
   - If the most suitable link belongs to a peer device in the RIR collaboration device group, the device forwards the packet to the peer device.

After finishing link selection, the device associates the quintuple of the packet with the most suitable link and records the association as the optimal link information for the session. The device forwards the subsequent packets of the same session based on the optimal link information. If no traffic is received for the session for a period of time, the device will delete the optimal link information.

Figure 9 RIR link selection workflow


Preference-based primary link selection
Is the optimal link found?
Yes
No
Preference-based backup link selection
Is the optimal link found?
Yes
No
Quality tolerant link selection
Is the optimal link found?
Yes
No
Bandwidth tolerant link selection
Return the link selection result
Start
Select a flow template based on the packet flow ID
Find the optimal link among the links in the flow template
Is the optimal link found?
No
Perform routing table-based forwarding
Yes
Forward the packet based on the link selection result
Link selection workflow

Preference-based primary link selection

RIR preferentially selects primary links that meet both the quality and bandwidth requirements for a service flow that matches a flow template. As shown in Figure 10, the device selects a primary link from the primary links in the flow template by examining the links in descending order of link preference. The device uses the following process to examine links with the same preference:

1. The device examines all links with the preference and identifies whether a link forms ECMP routes with other links. If a link forms ECMP routes with other links, the device further identifies whether the link is a primary link that meets both the quality and bandwidth requirements of the service.
    - If yes, the device adds the link to the available suitable link list of that preference.
    - If no, the device further identifies whether the link is a primary link that meets the bandwidth requirements of the service.
        - If yes, the device adds the link to the quality tolerant link list for quality tolerant link selection. Then, the device continues to examine other links with the same preference.
        - If no, the device continues to examine other links with the same preference.

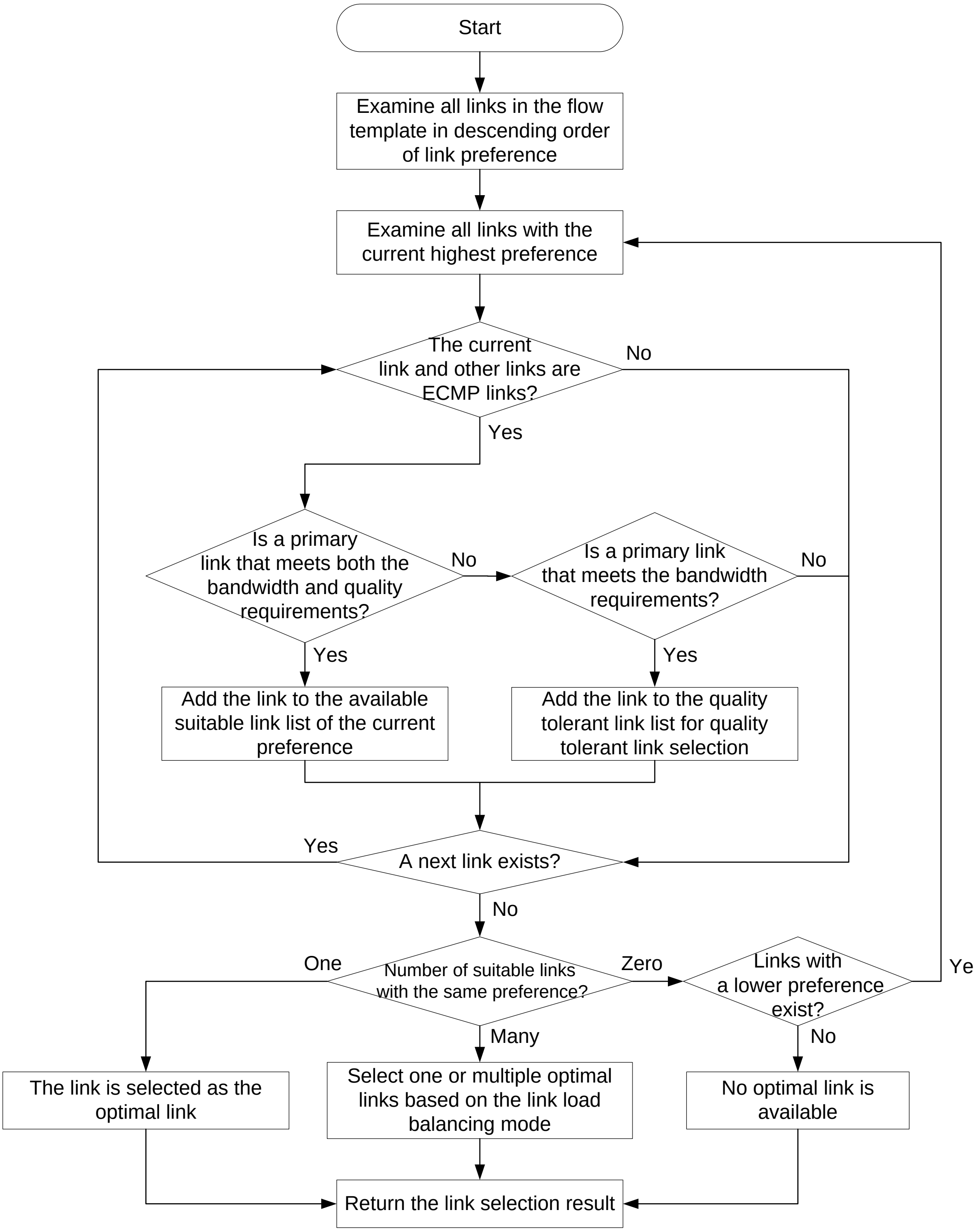
    If a link does not form ECMP routes with other links, the device continues to examine other links with the same preference.
2. When the device finishes examining all links with the preference, it identifies how many suitable links are available for the service flow.

- If only one suitable link is available, the device selects that link as the optimal link.
- If multiple suitable links are available, the device selects one or multiple optimal links from them based on the link load balancing mode. In a per-session load balancing mode, the device selects only one link as the optimal link of a session. In the per-packet load balancing mode, the device can select multiple links as the optimal links of a session.
- If no suitable link is available, the device examines the links that have a preference value lower than the links with the current preference.

If no primary links in the flow template are suitable, the device determines that no optimal primary link is found for the service flow.

For more information about identifying whether a link meets the quality requirements, see "Quality-based link selection." For more information about identifying whether a link meets the bandwidth requirements, see "Bandwidth-based link selection."

Figure 10 Preference-based primary link selection workflow


Start
Examine all links in the flow template in descending order of link preference
Examine all links with the current highest preference
The current link and other links are ECMP links?
No
Yes
Is a primary link that meets both the bandwidth and quality requirements?
No
Is a primary link that meets the bandwidth requirements?
No
Yes
Yes
Add the link to the available suitable link list of the current preference
Add the link to the quality tolerant link list for quality tolerant link selection
Yes
A next link exists?
No
One
Number of suitable links with the same preference?
Zero
Links with a lower preference exist?
Yes
Many
No
The link is selected as the optimal link
Select one or multiple optimal links based on the link load balancing mode
No optimal link is available
Return the link selection result

Preference-based backup link selection

If no primary link is suitable for a service flow, the device tries to find a backup link that meets both the quality and bandwidth requirements for the flow in the flow template. The link selection process is the same as that for selecting a primary link.

Quality tolerant link selection

If preference-based link selection fails to select a suitable link from both primary and backup links, the device performs quality tolerant link selection.

The links that meet the quality tolerant link selection criterion are those added to the quality tolerant link list during preference-based primary and backup link selection. These links do not meet the quality requirements of the service, but they meet the bandwidth requirements of the service. Quality tolerant link selection selects a link from the links that meet only the bandwidth requirements of the service.

If multiple quality tolerant links are available, the device selects one or multiple optimal links from them based on the link load balancing mode.

Bandwidth tolerant link selection

If quality tolerant link selection still cannot find a suitable link for a service flow, the device performs bandwidth tolerant link selection. Bandwidth tolerant link selection selects one link from ECMP routes in the flow template as the optimal link.

If multiple links are available, the device selects one or multiple optimal links from them based on the link load balancing mode.

Restrictions: Hardware compatibility with RIR

Models	RIR compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB1180, NFNX3-HDB1480	No

Restrictions and guidelines: RIR configuration

When a link has a large amount of traffic, do not change the link selection policy for that link. For example, change the link preference. If you change the link selection policy, the system might fail to perform link selection as expected. As a best practice, change the link selection policy for a link when the link does not have traffic or when the link has a small amount of traffic.

RIR tasks at a glance

Spoke configuration tasks at a glance

To configure RIR on a spoke, perform the following tasks:

1. Enabling the RIR process
2. Configuring a spoke
   - Enabling the RIR client
   - Specifying an RIR client synchronization port
   - Enabling the NQA client
   - (Optional.) Configuring NQA link connectivity probe parameters

- Configuring an NQA link quality operation
- Configuring an SLA

3. Configuring link attributes
   - Assigning a link type and index to a VSI interface
   - Configuring the link bandwidth of a VXLAN tunnel interface
   - (Optional.) Configuring RIR backup links
4. Configuring a flow template
   - Creating a flow template
   - Configuring a quality policy for the flow template
   - (Optional.) Specifying the per-session expected bandwidth
   - Specifying link preference values for links
5. (Optional.) Configuring the link load balancing mode
   - Setting the per-session periodic link adjustment mode
   - Setting the per-packet load balancing mode

   By default, the per-session weight-based link selection mode is used.
6. (Optional.) Configuring flow priority-based traffic scheduling
7. (Optional.) Setting the link selection delay and link selection suppression period
8. Configuring a QoS policy to mark matching packets with a flow ID
   a. Creating a traffic class and defining packet match criteria
   b. Creating a traffic behavior and configuring a flow ID marking action
   c. Configuring a QoS policy
   d. Applying the QoS policy to an interface
9. (Optional.) Configuring RIR collaboration
   a. Setting up RIR dedicated links between local and peer devices
   b. Applying QoS policies to interfaces interconnecting local and peer devices
   c. Assigning links to an RIR collaboration link group
   d. Establishing RIR collaboration relationship for each pair of local and peer devices
   e. Configuring RIR packet redirection
10. (Optional.) Enabling RIR logging
11. (Optional.) Configuring flow ID-based traffic rate statistics for tunnels

Hub configuration tasks at a glance

To configure RIR on a hub, perform the following tasks:

1. Enabling the RIR process
2. Configuring a hub
   - Enabling the RIR server
   - Specifying an RIR server synchronization port
   - Configuring the NQA server
   - (Optional.) Creating an SLA and an NQA link quality operation
3. Configuring link attributes
   - Assigning a link type and index to a VSI interface
   - Configuring the link bandwidth of a VXLAN tunnel interface
   - (Optional.) Configuring RIR backup links

4. Configuring a flow template
   - Creating a flow template
   - Configuring a quality policy for the flow template
   - (Optional.) Specifying the per-session expected bandwidth
   - Specifying link preference values for links
5. (Optional.) Configuring the link load balancing mode
   - Setting the per-session periodic link adjustment mode
   - Setting the per-packet load balancing mode

   By default, the per-session weight-based link selection mode is used.
6. (Optional.) Configuring flow priority-based traffic scheduling
7. (Optional.) Setting the link selection delay and link selection suppression period
8. Configuring a QoS policy to mark matching packets with a flow ID
   a. Creating a traffic class and defining packet match criteria
   b. Creating a traffic behavior and configuring a flow ID marking action
   c. Configuring a QoS policy
   d. Applying the QoS policy to an interface
9. (Optional.) Configuring RIR collaboration
   a. Setting up RIR dedicated links between local and peer devices
   b. Applying QoS policies to interfaces interconnecting local and peer devices
   c. Assigning links to an RIR collaboration link group
   d. Establishing RIR collaboration relationship for each pair of local and peer devices
   e. Configuring RIR packet redirection
10. (Optional.) Enabling RIR logging
11. (Optional.) Configuring flow ID-based traffic rate statistics for tunnels

Prerequisites for RIR configuration

Make sure the hub and each spoke have a minimum of two ECMP routes to reach each other.

Enabling the RIR process

1. Enter system view.
   ```
   system-view
   ```
2. Enable the RIR process and enter RIR view.
   ```
   rir
   ```
   By default, the RIR process is disabled.

Configuring a spoke

Enabling the RIR client

About this task

To avoid NQA probes from occupying too many resources on a hub in a hub-spoke network, configure the hub as an RIR server and configure the spokes as RIR clients.

You can enable the RIR client globally or on an interface.

- Enabling the RIR client globally also enables the RIR client for all interfaces on the device. The interfaces can send link quality probe results for the RIR client.
- Enabling the RIR client on an interface allows only that interface to send link quality probe results for the RIR client.

Restrictions and guidelines

When you enable the RIR client, follow these restrictions and guidelines:

- In a VXLAN network, only tunnel interfaces support enabling the RIR client. The RIR client uses the tunnel interfaces to send link quality probe results.
- The RIR server and RIR client cannot be both enabled on the same interface.
- If the enabled role (RIR server or client) on an interface is different from the globally enabled role, the interface-specific role takes effect on that interface.

Procedure

1. Enter system view.

   system-view

2. Enable the RIR client. Perform the following tasks as needed:
   - Execute the following commands in sequence to enable the RIR client globally:

     rir

     client enable
   - Execute the following commands in sequence to enable the RIR client on a VXLAN tunnel interface:

     interface tunnel *tunnel-number*

     rir role client

   By default, the RIR client is disabled globally and on an interface.

Specifying an RIR client synchronization port

About this task

Perform this task to specify a port for an RIR client to synchronize link quality probe results to RIR servers.

Restrictions and guidelines

Specify the same synchronization port on the RIR client and server for successful synchronization of link quality probe results.

Procedure

1. Enter system view.

   system-view

2. Enter RIR view.

   rir

3. Specify a port for the RIR client to synchronize probe information to RIR servers.

   probe sync-port *port-number*

   By default, no port is specified for an RIR client to synchronize probe information to RIR servers.

Enabling the NQA client

About this task

An RIR client also acts as an NQA client. You must enable the NQA client on a spoke to ensure that NQA link connectivity probes and link quality probes can be performed correctly.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the NQA client.

   ```
   nqa agent enable
   ```

   By default, the NQA client is disabled.

   For more information about this command, see NQA commands in *Network Management and Monitoring Command Reference*.

Configuring NQA link connectivity probe parameters

About this task

An NQA client starts NQA link connectivity probes on all links configured on flow templates after RIR is enabled. A spoke (NQA client) performs consecutive probes at intervals as configured and waits for responses for the packets. If the client does not receive any responses when the probe packet timeout timer expires, a link connectivity issue exists.

In a VXLAN network, the NQA link connectivity probe targets are the VXLAN tunnel interfaces enabled with the RIR client.

A link connectivity probe packet uses the source IP address of a VXLAN tunnel as its source IP address and uses the tunnel destination IP address as its destination IP address.

Restrictions and guidelines

Setting a shorter probe interval obtains more precise probe results but requires more system resources.

Set a shorter probe packet timeout time if the requirement for link quality is high.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIR view.

   ```
   rir
   ```

3. Configure NQA link connectivity probe parameters.

   ```
   probe connect interval interval timeout timeout
   ```

   By default, the NQA probe interval is 100 milliseconds and the NQA packet timeout time is 3000 milliseconds.

Configuring an NQA link quality operation

About this task

An NQA link quality operation allows a flow template to start UDP jitter probes based on the probe parameters in the operation in order to detect the quality of links.

You can configure a quality policy for a flow template to associate the flow template with an SLA and an NQA link quality operation. The device monitors the quality of links in the flow template based on the NQA link quality operation and compares the NQA probe results with the thresholds in the SLA. If all parameter values in the probe results of a link are lower than or equal to the thresholds in the SLA, the link is qualified for the flow.

To differentiate service flows that have different link quality requirements, associate the flow templates with NQA link quality operations that contain different probe parameter values. Two NQA link quality operations with different probe parameter values might offer different probe results for the same link.

In a VXLAN network, the NQA link quality probe targets are the VXLAN tunnel interfaces enabled with the RIR client.

A link quality probe packet uses the source IP address of a VXLAN tunnel as its source IP address and uses the tunnel destination IP address as its destination IP address.

Restrictions and guidelines

NQA link quality probes are used in conjunction with the NQA server and client features. For a spoke (NQA client) to perform NQA link quality probes, make sure UDP listening services have been configured on the NQA server.

Procedure

1. Enter system view.
   system-view
2. Enter RIR view.
   rir
3. Create an NQA link quality operation and enter its view.
   nqa *nqa-id*
4. Configure NQA link quality probe parameters.
   - Set the intervals at which the NQA client performs consecutive probes.
     probe interval *interval*

     By default, the NQA link quality probe interval is 100 milliseconds.
   - Set the DSCP value of NQA link quality probe packets.
     probe packet-dscp *dscp-value*

     By default, the DSCP value of NQA link quality probe packets is 63.
   - Set the number of NQA link quality probe packets sent per probe.
     probe packet-number *number*

     By default, 100 NQA link quality probe packets are sent per probe.
   - Set the intervals at which NQA link quality probe packets are sent.
     probe packet-interval *interval*

     By default, NQA link quality probe packets are sent at intervals of 20 milliseconds.
   - Set the timeout time for waiting for a response to an NQA link quality probe packet.
     probe packet-timeout *packet-timeout*

     By default, the timeout time is 3000 milliseconds.
   - Specify a destination port for NQA link quality probes.
     probe port *port-number*

     By default, no destination port is specified for NQA link quality probes.

     To correctly perform NQA link quality probes, the destination port number must be the same as the listening port on the NQA server.

Configuring an SLA

About this task

You can specify an SLA and NQA link quality operation for a flow template. The device monitors the quality of links in the flow template based on the NQA link quality operation and compares the NQA probe results with the thresholds in the SLA. The device selects only links that meet the quality requirements of the SLA for traffic that matches the flow template.

Perform this task to create an SLA and configure its link quality thresholds. Two SLAs might offer different quality results for the same NQA link quality operation.

Procedure

1. Enter system view.
   `system-view`
2. Enter RIR view.
   `rir`
3. Create an SLA and enter its view.
   `sla` *`sla-id`*
4. Configure NQA link quality thresholds.
   - Configure the link delay threshold.
     `delay threshold` *`threshold-value`*
     By default, the link delay threshold is 10 milliseconds.
   - Configure the link jitter threshold.
     `jitter threshold` *`threshold-value`*
     By default, the link jitter threshold is 100 milliseconds.
   - Configure the packet loss threshold.
     `packet-loss threshold` *`threshold-value`*
     By default, the packet loss threshold is 100‰.

Configuring a hub

Enabling the RIR server

About this task

To avoid NQA probes from occupying too many resources on a hub in a hub-spoke network, configure the hub as an RIR server and configure the spokes as RIR clients.

You can enable the RIR server globally or on an interface.

- Enabling the RIR server globally also enables the RIR server for all interfaces on the device. The interfaces can receive link quality probe results synchronized from RIR clients.
- Enabling the RIR server on an interface allows only that interface to receive link quality probe results synchronized from RIR clients.

Restrictions and guidelines

When you enable the RIR server, follow these restrictions and guidelines:

- In a VXLAN network, only tunnel interfaces support enabling the RIR server. The RIR server uses the tunnel interfaces to receive link quality probe results synchronized from RIR clients.
- The RIR server and RIR client cannot be both enabled on the same interface.

- If the enabled role (RIR server or client) on an interface is different from the globally enabled role, the interface-specific role takes effect on that interface.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enable the RIR server. Perform the following tasks as needed:
   - Execute the following commands in sequence to enable the RIR server globally:
     ```
     rir
     server enable
     ```
   - Execute the following commands in sequence to enable the RIR server on a VXLAN tunnel interface:
     ```
     interface tunnel tunnel-number
     rir role server
     ```

   By default, the RIR server is disabled globally and on an interface.

Specifying an RIR server synchronization port

About this task

Perform this task to specify a port for an RIR server to receive link quality probe results synchronized from RIR clients.

Restrictions and guidelines

Specify the same synchronization port on the RIR client and server for successful synchronization of link quality probe results.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enter RIR view.
   ```
   rir
   ```
3. Specify a port for receiving probe information synchronized from RIR clients.
   ```
   probe sync-port port-number
   ```
   By default, no port is specified for receiving probe information synchronized from RIR clients.

Configuring the NQA server

About this task

A hub that acts as an RIR server also acts as an NQA server. To ensure that NQA link connectivity probes can be performed correctly, you must enable the NQA server on the hub. To ensure that NQA link quality probes can be performed correctly, configure UDP listening services.

Restrictions and guidelines

The listening port specified on the hub (NQA server) must be the same as the destination port number specified in NQA link quality operations on the spokes (NQA clients). In addition, make sure the port is not used by any other services.

Procedure

1. Enter system view.

```
system-view
```

2. Enable the NQA server.

   ```
   nqa server enable
   ```

   By default, the NQA server is disabled.

   For more information about this command, see NQA commands in *Network Management and Monitoring Command Reference*.

3. Configure a UDP listening service for the NQA server to listen to a port on an IP address.

   ```
   nqa server udp-echo ip-address port-number [ vpn-instance
   vpn-instance-name ] [ tos tos ]
   ```

   For more information about this command, see NQA commands in *Network Management and Monitoring Command Reference*.

Creating an SLA and an NQA link quality operation

About this task

When you configure the quality policy of a flow template on a hub, you must specify an SLA and an NQA link quality operation. Perform this task to create the SLA and NQA link quality operation specified in the quality policy. On the hub, you do not need to configure the parameters in the SLA and NQA link quality operation. If you configure these parameters, the configuration does not take effect. For more information about quality policy configuration, see "Configuring a quality policy for the flow template."

Restrictions and guidelines

If you do not plan to configure the quality policy of a flow template on a hub, you do not need to create an SLA and NQA link quality operation.

The hub and spoke can have different SLA and NQA link quality operation settings in the quality policy of the same flow template. The difference does not affect the execution and application of the quality policy. As a best practice to identify the quality policy, configure the same SLA and NQA link quality operation on the hub and spoke.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter RIR view.

   ```
   rir
   ```

3. Create an SLA and enter its view.

   ```
   sla sla-id
   ```

4. Return to RIR view.

   ```
   quit
   ```

5. Create an NQA link quality operation and enter its view.

   ```
   nqa nqa-id
   ```

Configuring link attributes

Assigning a link type and index to a VSI interface

About this task

The link type and link index together uniquely identify a link between a hub and a spoke. For a flow template to use a link, you must assign a link type and index to the link. Perform this task to configure the link type as 4G, Internet, MPLS, or MSTP. The link type only marks the network type of the link and it does not affect packet encapsulation.

VXLAN-based RIR allows a hub and a spoke to have only one VXLAN tunnel for a VSI interface (a VXLAN). By assigning a link type and index to the VSI interface, RIR can identify the VXLAN tunnel between the hub and spoke.

A VSI interface on a hub (or spoke) can have a VXLAN tunnel to each spoke (or hub). The VXLAN tunnels of the same VSI interface are assigned the same link type and link index.

Restrictions and guidelines

Only 4G, Internet, MPLS, and MSTP link types are supported.

The link type is used only for identifying links, and it does not affect packet encapsulation.

A VSI interface can be associated only with one link type.

You must assign different link indexes to the same type of links on different VSI interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter VSI interface view.

   interface vsi-interface *vsi-interface-id*

3. Assign a link type and index to the VSI interface.

   rir link-type { **4g** | **internet** | **mpls** | **mstp** } **index** *link-index*

   By default, no link type or index is assigned to a VSI interface.

Configuring the link bandwidth of a VXLAN tunnel interface

1. Enter system view.

   system-view

2. Enter VXLAN tunnel interface view.

   interface tunnel *tunnel-number*

3. Configure the expected link bandwidth of the VXLAN tunnel interface.

   bandwidth *bandwidth-value*

   The default expected bandwidth (in kbps) is the interface maximum rate divided by 1000.

   The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

   For more information about this command, see tunneling commands in *Layer 3—IP Services Command Reference*.

Configuring RIR backup links

About this task

The links between a spoke and a primary hub are typically primary links, and the links between a spoke and a backup hub are typically backup links. RIR selects qualified primary links prior to qualified backup links.

In a VXLAN network, the links (VXLAN tunnels) that are assigned link preference values are primary links by default. If you configure a VXLAN tunnel as an RIR backup tunnel, RIR uses the VXLAN tunnel as an RIR backup link.

Restrictions and guidelines

Configure VXLAN tunnels as backup links depending on the network requirements. You can configure a VXLAN tunnel between a spoke and a primary hub as a backup link.

Procedure

1. Enter system view.

   `system-view`

2. Enter VXLAN tunnel interface view.

   `interface tunnel` *`number`*

3. Configure the tunnel as an RIR backup tunnel.

   `rir backup`

   By default, a tunnel is an RIR primary tunnel.

Configuring a flow template

About flow template configuration

Configure a flow template to determine the link selection policies for a type of service flow.

Creating a flow template

About this task

To define link selection policies for a type of service flow, you can create a flow template and configure link selection policies in the flow template. By marking the type of service flow with the flow ID of the flow template, the device can use the link selection policies in the flow template to select links for that type of service flow.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIR view.

   `rir`

3. Create a flow template and enter its view.

   `flow` *`flow-id`*

Configuring a quality policy for the flow template

About this task

If you configure the quality policy of a flow template on a hub, the hub performs link quality detection based on the link quality probe results synchronized from spokes. If you do not configure the quality policy on the spokes for the same flow template, the hub cannot obtain link quality probe results from any spokes. Because the hub does not perform link quality probe on its own, it determines that all links in the flow template fail quality detection.

If you do not configure the quality policy of a flow template on a hub, the hub determines that all links in the flow template meet the service quality requirements.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIR view.

   `rir`

3. Enter flow template view.

   `flow` *`flow-id`*

4. Configure a quality policy for the flow template.

   `quality-policy sla` *`sla-id`* **`nqa`** *`nqa-id`*

   By default, no quality policy is configured for a flow template.

   The specified SLA and NQA link quality operation must already exist.

Specifying the per-session expected bandwidth

About this task

To select a link for traffic of a session, a device first performs bandwidth detection based on the per-session expected bandwidth in the flow template to which the session belongs. If the used bandwidth plus the per-session expected bandwidth of a candidate link is less than 80% of its total bandwidth, the current available bandwidth of the candidate link meets the session bandwidth requirements. The link passes the bandwidth detection.

The per-session expected bandwidth is not the actual bandwidth of a session. It is only a value estimated based on the user services.

Procedure

1. Enter system view.

   `system-view`

2. Enter RIR view.

   `rir`

3. Enter flow template view.

   `flow` *`flow-id`*

4. Specify the per-session expected bandwidth.

   `expect-bandwidth` *`bandwidth`*

   By default, the per-session expected bandwidth is 0 kbps.

Specifying link preference values for links

About this task

RIR preferentially selects links with higher preference for a type of service flow.

VXLAN-based RIR supports assigning a link preference to a type of links with a specific link index in flow template view. The link type and link index identify links on a VSI interface. As VXLAN-based RIR allows a hub and spoke to have only one VXLAN tunnel for a VSI interface, the link preference configured on the VSI interface is the link preference of the VXLAN tunnel between the hub and spoke.

Restrictions and guidelines

You can assign the same link preference value to links with different link types and indexes in the same flow template.

Procedure

1. Enter system view.

   system-view

2. Enter RIR view.

   rir

3. Enter flow template view.

   flow *flow-id*

4. Configure a link preference for the specified type of links with the specified link index.

   path link-type { **4g** | **internet** | **mpls** | **mstp** } **index** *link-index* **preference** *preference*

   By default, no link preference is configured for a type of links with a specific link index in a flow template.

Configuring the link load balancing mode

Restrictions and guidelines for link load balancing mode configuration

For a flow template, the per-packet load balancing mode takes precedence over the global per-session periodic link adjustment mode. If the per-packet load balancing mode is not enabled for a flow template, the flow template uses the global link load balancing mode.

Setting the per-session periodic link adjustment mode

1. Enter system view.

   system-view

2. Enter RIR view.

   rir

3. Set the link load balancing mode to the per-session periodic link adjustment mode.

   load-balance per-session periodic-adjust enable

   By default, the link load balancing mode is per-session weight-based link selection mode.

4. Set the adjustment interval for the per-session periodic link adjustment mode.

load-balance per-session periodic-adjust adjust-interval *interval-value*

By default, the adjustment interval for the per-session periodic link adjustment mode is 30 seconds.

5. Set the periodic adjustment thresholds in per-session periodic link adjustment mode.

   load-balance per-session periodic-adjust threshold upper *upper-threshold-value* **lower** *lower-threshold-value*

   By default, the periodic adjustment upper threshold is 50% and the periodic adjustment lower threshold is 20%.

   The periodic adjustment upper threshold must be greater than or equal to the periodic adjustment lower threshold.

Setting the per-packet load balancing mode

Restrictions and guidelines

Because packets of the same session are distributed to multiple links, the receiver might receive out-of-order packets. As a best practice, do not enable per-packet load balancing for order-sensitive services (except the services that use protocols to maintain a correct packet order, for example, TCP).

Procedure

1. Enter system view.

   system-view

2. Enter RIR view.

   rir

3. Enter flow template view.

   flow *flow-id*

4. Set the link load balancing mode to the per-packet mode.

   load-balance per-packet enable

   By default, the RIR global link load balancing mode applies.

Configuring flow priority-based traffic scheduling

About this task

To ensure that services with higher priority preferentially use link resources, enable flow priority-based traffic scheduling.

The priority of a flow that matches a flow template is determined by the ID of the SLA associated with that flow template. The greater the SLA ID is, the higher the flow priority. To specify an SLA for a flow template, use the **quality-policy** command. If the command is not configured in a flow template, flows that match the flow template have the lowest priority.

If flow priority-based traffic scheduling is enabled, traffic scheduling is triggered when the bandwidth usage of a link exceeds the upper threshold. The scheduling might be last for several scheduling periods. Within each scheduling period, RIR redistributes the current lowest priority flow on this link to other links. The scheduling does not stop until the bandwidth usage of all links for the current lowest priority flow is below the lower threshold or only the highest priority flow is left on this link.

Procedure

1. Enter system view.

   system-view

2. Enter RIR view.

   `rir`

3. Enable flow priority-based traffic scheduling.

   `flow priority-based-schedule enable`

   By default, flow priority-based traffic scheduling is disabled.

4. Set the scheduling period for flow priority-based traffic scheduling.

   `flow priority-based-schedule schedule-period` *`schedule-period-value`*

   By default, the scheduling period for flow priority-based traffic scheduling is 30 seconds.

5. Set the bandwidth usage thresholds for flow priority-based traffic scheduling.

   `flow priority-based-schedule bandwidth-threshold upper` *`upper-threshold`* **`lower`** *`lower-threshold`*

   By default, the bandwidth usage upper threshold is 90% and the bandwidth usage lower threshold is 20%.

   The upper threshold must be greater than or equal to the lower threshold.

Setting the link selection delay and link selection suppression period

1. Enter system view.

   `system-view`

2. Enter RIR view.

   `rir`

3. Set the link selection delay.

   `link-select delay` *`delay`*

   By default, the link selection delay is 60 seconds.

4. Set the link selection suppression period.

   `link-select suppress-period` *`period-value`*

   By default, no link selection suppression period is configured. The device does not start the link selection suppression period after a link selection.

   As a best practice, set the link selection suppression period to a multiple of the link selection delay time. Make sure the suppression period is at least double of the link selection delay time.

Configuring a QoS policy to mark matching packets with a flow ID

About configuring a QoS policy to mark matching packets with a flow ID

Apply a QoS policy to an interface to mark matching packets on the interface with a flow ID. RIR processes packets marked with a flow ID based on the flow template that uses the flow ID. For more information about QoS marking, see QoS overview, QoS policies, and marking configuration in *ACL and QoS Configuration Guide*.

Creating a traffic class and defining packet match criteria

1. Enter system view.

   system-view

2. Create a traffic class and enter its view.

   traffic classifier *classifier-name* [**operator** { **and** | **or** }]

3. Define packet match criteria, including the quintuple and DSCP.

   if-match [**not**] *match-criteria*

   By default, no packet match criteria are defined.

Creating a traffic behavior and configuring a flow ID marking action

1. Enter system view.

   system-view

2. Create a traffic behavior and enter its view.

   traffic behavior *behavior-name*

3. Configure the traffic behavior to mark matching traffic with the specified flow ID.

   remark flow-id *flow-id*

   By default, no flow ID marking action is configured.

Configuring a QoS policy

1. Enter system view.

   system-view

2. Create a QoS policy and enter its view.

   qos policy *policy-name*

3. Associate the traffic behavior with the traffic class in the QoS policy.

   classifier *classifier-name* **behavior** *behavior-name*

   By default, no traffic behavior is associated with a traffic class.

Applying the QoS policy to an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Apply the QoS policy to the inbound direction of the interface.

   qos apply policy *policy-name* **inbound**

   By default, no QoS policy is applied to an interface.

Configuring RIR collaboration

Restrictions and guidelines for RIR collaboration configuration

All devices that form an RIR collaboration device group must have independent RIR capabilities. They must mark the same type of service traffic with the same flow ID.

Perform the RIR collaboration tasks in this section on both the local and peer devices in an RIR collaboration device group.

Setting up RIR dedicated links between local and peer devices

Use RIR dedicated links to ensure that packets can be correctly forwarded between local and peer devices. You can configure direct routes or GRE tunnels as dedicated links as follows:

- For directly-connected peer devices, you can configure multiple physical interfaces or subinterfaces on the devices. The interfaces or subinterfaces will generate direct routes for packets in the public network and each VPN instance. For more information, see Ethernet interface configuration in *Interface Configuration Guide*.
- For indirectly-connected peer devices, manually configure GRE tunnels between them for packets in the public network and each VPN instance. For more information about GRE configuration, see *Layer 3—IP Services Configuration Guide*.

Applying QoS policies to interfaces interconnecting local and peer devices

Outgoing packets do not carry the flow ID marked in the RIR process. To mark a flow ID for received packets, apply a QoS policy to the interfaces interconnecting the local and peer devices. For more information, see "Configuring a QoS policy to mark matching packets with a flow ID."

Assigning links to an RIR collaboration link group

About this task

The local device can discover and select links synchronized from peer devices only after the links are assigned to an RIR collaboration link group. In the RIR collaboration link group, both local links and links synchronized from peer devices have the same destination.

Restrictions and guidelines

In an RIR collaboration device group, make sure all links to the same device or RIR collaboration device group are assigned to the same RIR collaboration link group.

In an RIR collaboration device group, make sure the links to different devices or RIR collaboration device groups are assigned to different RIR collaboration link groups.

In different RIR collaboration device groups, the links to the same device or RIR collaboration device group can be assigned to the same RIR collaboration link group. As a best practice to identify links, assign the links to different RIR collaboration link groups.

Procedure

1. Enter system view.

```
system-view
```

2. Enter VXLAN tunnel interface view.

```
interface tunnel number
```

3. Assign the VXLAN tunnel to an RIR collaboration link group.

```
rir collaboration-link-group group-id
```

By default, a VXLAN tunnel belongs to RIR collaboration link group 0.

Establishing RIR collaboration relationship for each pair of local and peer devices

About this task

Each pair of devices in an RIR collaboration device group must establish RIR collaboration relationship with each other for link data synchronization. After a pair of devices establish RIR collaboration relationship, the device with a lower IP address acts as the client to initiate a TCP connection to the other device. Through the TCP connection, the local device can synchronize the configuration and status data of links that meet the service requirements to the peer device. The data does not include the link data synchronized from other devices in the same RIR collaboration device group.

Restrictions and guidelines

The local and peer IP addresses used to establish RIR collaboration relationship must belong to the public network or the same VPN instance.

A pair of RIR collaboration devices can establish only one TCP connection to synchronize link data.

Procedure

1. Enter system view.

```
system-view
```

2. Enter RIR view.

```
rir
```

3. Enable the local device to establish RIR collaboration relationship with a peer device.

```
collaboration peer [ vpn-instance vpn-instance-name ]
peer-ipv4-address local local-ipv4-address sync-port port-number
```

By default, the local device does not establish RIR collaboration relationship with any device.

Configuring RIR packet redirection

About this task

For the local device to select links from a peer device, you must configure RIR packet redirection. Perform this task to specify the redirect IP address for packets redirected to the peer device on the public network or a VPN instance. When the local device selects links from the peer device to forward packets on the public network or a VPN instance, it performs the following operations:

1. Looks up the routing table of the public network or VPN instance based on the redirect IP address.
2. Forwards the packets to the peer device through the RIR dedicated link.

Procedure

1. Enter system view.

```
system-view
```

2. Enter RIR view.

rir

3. Specify the IP address to which the local device redirects packets forwarded to an RIR collaboration peer.

   collaboration peer [**vpn-instance** *vpn-instance-name*] **peer-ipv4-address redirect** [**vpn-instance** *redirect-vpn-instance-name*] *redirect-ipv4-address*

   By default, no IP address is specified for the local device to redirect packets forwarded to an RIR collaboration peer.

   The **vpn-instance** *vpn-instance-name* option specifies the VPN instance on which the local and peer devices establish RIR collaboration relationship. The **vpn-instance** *redirect-vpn-instance-name* option specifies a redirect VPN instance. When the local device selects a link from a peer device for forwarding traffic in the redirect VPN instance, the local device redirects the packets to the redirect IPv4 address.

Enabling RIR logging

About this task

RIR logs record events occurred during the RIR process, such as link selection and reselection, quality change, bandwidth change, configuration change, and link fault events. The logs help the administrator analyze, maintain, and adjust the RIR network.

RIR logs are flow logs. To output RIR logs, you must also configure flow log features. For more information about flow logs, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter RIR view.

   rir

3. Enable RIR logging.

   log enable

   By default, RIR logging is disabled.

Configuring flow ID-based traffic rate statistics for tunnels

About this task

This feature enables the device to periodically collect traffic rate statistics for tunnels on a flow ID basis.

Procedure

1. Enter system view.

   system-view

2. Enable flow ID-based traffic rate statistics for tunnels.

   tunnel flow-statistics enable

   By default, flow ID-based traffic rate statistics for tunnels is disabled.

3. (Optional.) Set the intervals at which the device collects flow ID-based traffic rate statistics for tunnels.

tunnel flow-statistics interval *interval*

By default, the device collects flow ID-based traffic rate statistics for tunnels at intervals of 300 seconds.

Display and maintenance commands for RIR

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display flow ID-based traffic rate statistics for tunnels.	**display tunnel flow-statistics** [**flow** *flow-id* [**interface tunnel** *number*]]
Clear flow ID-based traffic rate statistics for tunnels.	**reset tunnel flow-statistics** [**flow** *flow-id* [**interface tunnel** *number*]]

NSFOCUS Firewall Series

NF ACL and QoS

Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for ACL and QoS features, including ACL, QoS, and time range.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

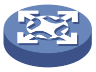
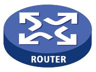
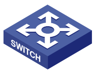
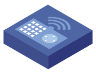
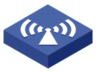
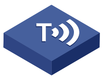
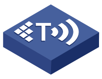
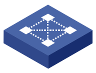
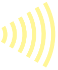
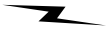
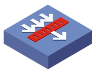
GUI conventions

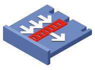
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring ACLs

About ACLs

An access control list (ACL) is a set of rules for identifying traffic based on criteria such as source IP address, destination IP address, and port number. The rules are also called permit or deny statements.

ACLs are primarily used for packet filtering. You can also use ACLs in QoS, security, routing, and other modules for identifying traffic. The packet drop or forwarding decisions depend on the modules that use ACLs.

Numbering and naming ACLs

When creating an ACL, you must assign it a number or name for identification. You can specify an existing ACL by its number or name. Each ACL type has a unique range of ACL numbers.

For basic or advanced ACLs with the same number, you must use the **`ipv6`** keyword to distinguish them. For ACLs with the same name, you must use the **`ipv6`** and **`mac`** keywords to distinguish them.

ACL types

Type	ACL number	IP version	Match criteria
Basic ACLs	2000 to 2999	IPv4	Source IPv4 address.
		IPv6	Source IPv6 address.
Advanced ACLs	3000 to 3999	IPv4	Source IPv4 address, destination IPv4 address, packet priority, protocol number, and other Layer 3 and Layer 4 header fields.
		IPv6	Source IPv6 address, destination IPv6 address, packet priority, protocol number, and other Layer 3 and Layer 4 header fields.
Layer 2 ACLs	4000 to 4999	IPv4 and IPv6	Layer 2 header fields, such as source and destination MAC addresses, 802.1p priority, and link layer protocol type.

Match order

The rules in an ACL are sorted in a specific order. When a packet matches a rule, the device stops the match process and performs the action defined in the rule. If an ACL contains overlapping or conflicting rules, the matching result and action to take depend on the rule order.

The following ACL match orders are available:

- **config**—Sorts ACL rules in ascending order of rule ID. A rule with a lower ID is matched before a rule with a higher ID. If you use this method, check the rules and their order carefully.
- **auto**—Sorts ACL rules in depth-first order. Depth-first ordering makes sure any subset of a rule is always matched before the rule. Table 1 lists the sequence of tie breakers that depth-first ordering uses to sort rules for each type of ACL.

Table 1 Sort ACL rules in depth-first order

ACL type	Sequence of tie breakers
IPv4 basic ACL	**1.** VPN instance. **2.** More 0s in the source IPv4 address wildcard (more 0s means a narrower IPv4 address range). **3.** Rule configured earlier.
IPv4 advanced ACL	**1.** VPN instance. **2.** Specific protocol number. **3.** More 0s in the source IPv4 address wildcard mask. **4.** More 0s in the destination IPv4 address wildcard. **5.** Narrower TCP/UDP service port number range. **6.** Rule configured earlier.
IPv6 basic ACL	**1.** VPN instance. **2.** Longer prefix for the source IPv6 address (a longer prefix means a narrower IPv6 address range). **3.** Rule configured earlier.
IPv6 advanced ACL	**1.** VPN instance. **2.** Specific protocol number. **3.** Longer prefix for the source IPv6 address. **4.** Longer prefix for the destination IPv6 address. **5.** Narrower TCP/UDP service port number range. **6.** Rule configured earlier.
Layer 2 ACL	**1.** More 1s in the source MAC address mask (more 1s means a smaller MAC address). **2.** More 1s in the destination MAC address mask. **3.** Rule configured earlier.

A wildcard mask, also called an inverse mask, is a 32-bit binary number represented in dotted decimal notation. In contrast to a network mask, the 0 bits in a wildcard mask represent "do care" bits, and the 1 bits represent "don't care" bits. If the "do care" bits in an IP address are identical to the "do care" bits in an IP address criterion, the IP address matches the criterion. All "don't care" bits are ignored. The 0s and 1s in a wildcard mask can be noncontiguous. For example, 0.255.0.255 is a valid wildcard mask.

Rule numbering

ACL rules can be manually numbered or automatically numbered. This section describes how automatic ACL rule numbering works.

Rule numbering step

If you do not assign an ID to the rule you are creating, the system automatically assigns it a rule ID. The rule numbering step sets the increment by which the system automatically numbers rules. For example, the default ACL rule numbering step is 5. If you do not assign IDs to rules you are creating, they are automatically numbered 0, 5, 10, 15, and so on. The wider the numbering step, the more rules you can insert between two rules.

By introducing a gap between rules rather than contiguously numbering rules, you have the flexibility of inserting rules in an ACL. This feature is important for a config-order ACL, where ACL rules are matched in ascending order of rule ID.

The rule numbering step sets the increment by which the system numbers rules automatically. If you do not specify a rule ID when creating an ACL rule, the system automatically assigns it a rule ID. This rule ID is the nearest higher multiple of the numbering step to the current highest rule ID, starting from the start rule ID. For example, if the rule numbering step is 5 and the current highest rule ID is 12, the rule is numbered 15.

The wider the numbering step, the more rules you can insert between two rules. Whenever the step changes, the rules are renumbered, starting from the start rule ID. For example, if there are five rules numbered 0, 5, 9, 10, and 15, changing the step from 5 to 2 causes the rules to be renumbered 0, 2, 4, 6, and 8.

Automatic rule numbering and renumbering

The ID automatically assigned to an ACL rule takes the nearest higher multiple of the numbering step to the current highest rule ID, starting with 0.

For example, if the step is 5, and there are five rules numbered 0, 5, 9, 10, and 12, the newly defined rule is numbered 15. If the ACL does not contain a rule, the first rule is numbered 0.

Whenever the step changes, the rules are renumbered, starting from 0. For example, changing the step from 5 to 2 renumbers rules 5, 10, 13, and 15 as rules 0, 2, 4, and 6.

For an ACL of the match order **auto**, rules are sorted in depth-first order, and are renumbered based on the match order. For example, rules are in the match order of 0, 10, and 5. Changing the numbering step to 2 renumbers rules 0, 10, and 5 (not 0, 5, and 10) as rules 0, 2, 4

Fragment filtering with ACLs

Traditional packet filtering matches only first fragments of packets, and allows all subsequent non-first fragments to pass through. Attackers can fabricate non-first fragments to attack networks.

To avoid risks, the ACL feature is designed as follows:

- Filters all fragments by default, including non-first fragments.
- Allows for matching criteria modification for efficiency. For example, you can configure the ACL to filter only non-first fragments.

Restrictions and guidelines: ACL configuration

- If you create a numbered ACL, you can enter the view of the ACL by using the following commands:
  - **acl** [**ipv6**] **number** *acl-number*
  - **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } *acl-number*
- If you create a ACL by specifying both a number and a name, you can enter the view of the ACL by using the following commands:
  - **acl** [**ipv6**] **number** *acl-number* (only for basic and advanced ACLs)
  - **acl** [**ipv6**] **number** *acl-number* [**name** *acl-name*]
  - **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } **name** *acl-name*
- If you create a named ACL by using the **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } **name** *acl-name* command, you can enter the view of the ACL by using the following commands:
  - **acl** [**ipv6**] **name** *acl-name*
  - **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } **name** *acl-name*
- Matching packets are forwarded through slow forwarding if an ACL rule contains match criteria or has functions enabled in addition to the following match criteria and functions:
  - Source and destination IP addresses.
  - Source and destination ports.
  - Transport layer protocol.
  - ICMP or ICMPv6 message type, message code, and message name.
  - VPN instance.

  - Logging.
  - Time range.

  Slow forwarding requires packets to be sent to the control plane for forwarding entry calculation, which affects the device forwarding performance.
- As a best practice to ensure device performance during peak hours, do not modify an ACL and the address object group that references it.

ACL tasks at a glance

To configure an ACL, perform the following tasks:

- Configure ACLs according to the characteristics of the packets to be matched
  - Configuring a basic ACL
  - Configuring an advanced ACL
  - Configuring a Layer 2 ACL
- (Optional.) Copying an ACL
- (Optional.) Enabling ACL acceleration
- (Optional.) Configuring packet filtering with ACLs

Configuring a basic ACL

About basic ACLs

Basic ACLs match packets based only on source IP addresses.

Restrictions and guidelines for basic ACL configuration

The **logging** keyword specified in an ACL rule enables the ACL module to send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output ACL logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view ACL logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Configuring an IPv4 basic ACL

1. Enter system view.

   system-view
2. Create an IPv4 basic ACL and enter its view. Choose one option as needed:
   - Create an IPv4 basic ACL by specifying an ACL number.

     acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

- Create an IPv4 basic ACL by specifying the **basic** keyword.

  acl basic { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

3. (Optional.) Configure a description for the IPv4 basic ACL.

   description *text*

   By default, an IPv4 basic ACL does not have a description.
4. (Optional.) Enable rule ID preemption.

   rule insert-only enable

   By default, rule ID preemption is disabled.
5. (Optional.) Set the rule numbering step.

   step *step-value*

   By default, the rule numbering step is 5 and the start rule ID is 0.
6. Create or edit a rule.

   rule [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **source** { **object-group** *address-group-name* | *source-address source-wildcard* | **any** } | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *

   The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging.
7. (Optional.) Add or edit a rule comment.

   rule *rule-id* **comment** *text*

   By default, no rule comment is configured.

Configuring an IPv6 basic ACL

1. Enter system view.

   system-view
2. Create an IPv6 basic ACL view and enter its view. Choose one option as needed:
   - Create an IPv6 basic ACL by specifying an ACL number.

     acl ipv6 number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]
   - Create an IPv6 basic ACL by specifying the **basic** keyword.

     acl ipv6 basic { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]
3. (Optional.) Configure a description for the IPv6 basic ACL.

   description *text*

   By default, an IPv6 basic ACL does not have a description.
4. (Optional.) Enable rule ID preemption.

   rule insert-only enable

   By default, rule ID preemption is disabled.
5. (Optional.) Set the rule numbering step.

   step *step-value*

   By default, the rule numbering step is 5 and the start rule ID is 0.
6. Create or edit a rule.

   rule [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **routing** [**type** *routing-type*] | **source** { **object-group** *address-group-name* |

source-address source-prefix | *source-address/source-prefix* | **any** } | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *

The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging.

7. (Optional.) Add or edit a rule comment.

   rule *rule-id* **comment** *text*

   By default, no rule comment is configured.

Configuring an advanced ACL

About advanced ACLs

Advanced ACLs match packets based on the following criteria:

- Source IP addresses.
- Destination IP addresses.
- Packet priorities.
- Protocol types.
- Other protocol header information, such as TCP/UDP source and destination port numbers, TCP flags, ICMP message types, and ICMP message codes.

Compared to basic ACLs, advanced ACLs allow more flexible and accurate filtering.

Restrictions and guidelines for advanced ACL configuration

The **logging** keyword specified in an ACL rule enables the ACL module to send log messages to the information center.

With the information center, you can set log message filtering and output rules, including output destinations.

The information center can output ACL logs to any destinations except the console and the monitor terminal. If you configure the console or monitor terminal as an output destination, the output destination setting will not take effect.

To view ACL logs stored on the device, use the **display logbuffer** command. Make sure you do not disable log output to the log buffer, which is enabled by default.

For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

To prevent device performance from degrading, do not configure an advanced ACL rule to match the DSCP field.

Configuring an IPv4 advanced ACL

1. Enter system view.

   system-view

2. Create an IPv4 advanced ACL and enter its view. Choose one option as needed:
   - Create a numbered IPv4 advanced ACL by specifying an ACL number.

     acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]
   - Create an IPv4 advanced ACL by specifying the **advanced** keyword.

acl advanced { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

3. (Optional.) Configure a description for the IPv4 advanced ACL.

   description *text*

   By default, an IPv4 advanced ACL does not have a description.

4. (Optional.) Enable rule ID preemption.

   rule insert-only enable

   By default, rule ID preemption is disabled.

5. (Optional.) Set the rule numbering step.

   step *step-value*

   By default, the rule numbering step is 5 and the start rule ID is 0.

6. Creaete or edit a rule.

   rule [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { **object-group** *address-group-name* | *dest-address dest-wildcard* | **any** } | **destination-port** { **object-group** *port-group-name* | *operator port1* [*port2*] } | { **dscp** *dscp* | { **precedence** *precedence* | **tos** *tos* } * } | **fragment** | **icmp-type** { *icmp-type* [*icmp-code*] | *icmp-message* } | **logging** | **source** { **object-group** *address-group-name* | *source-address source-wildcard* | **any** } | **source-port** { **object-group** *port-group-name* | *operator port1* [*port2*] } | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *

   The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging.

7. (Optional.) Add or edit a rule comment.

   rule *rule-id* **comment** *text*

   By default, no rule comment is configured.

Configuring an IPv6 advanced ACL

1. Enter system view.

   system-view

2. Create an IPv6 advanced ACL and enter its view. Choose one option as needed:
   - Create a numbered IPv6 advanced ACL by specifying an ACL number.

     acl ipv6 number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]
   - Create an IPv6 advanced ACL by specifying the **advanced** keyword.

     acl ipv6 advanced { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

3. (Optional.) Configure a description for the IPv6 advanced ACL.

   description *text*

   By default, an IPv6 advanced ACL does not have a description.

4. (Optional.) Enable rule ID preemption.

   rule insert-only enable

   By default, rule ID preemption is disabled.

5. (Optional.) Set the rule numbering step.

step *step-value*

By default, the rule numbering step is 5 and the start rule ID is 0.

6. Create or edit a rule.

   rule [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { **object-group** *address-group-name* | *dest-address dest-prefix* | *dest-address/dest-prefix* | **any** } | **destination-port** { **object-group** *port-group-name* | *operator port1* [*port2*] } | **dscp** *dscp* | **flow-label** *flow-label-value* | **fragment** | **icmp6-type** { *icmp6-type icmp6-code* | *icmp6-message* } | **logging** | **routing** [**type** *routing-type*] | **hop-by-hop** [**type** *hop-type*] | **source** { **object-group** *address-group-name* | *source-address source-prefix* | *source-address/source-prefix* | **any** } | **source-port** { **object-group** *port-group-name* | *operator port1* [*port2*] } | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *

   The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging.

7. (Optional.) Add or edit a rule comment.

   rule *rule-id* **comment** *text*

   By default, no rule comment is configured.

Configuring a Layer 2 ACL

About this task

Layer 2 ACLs, also called Ethernet frame header ACLs, match packets based on Layer 2 Ethernet header fields, such as:

- Source MAC address.
- Destination MAC address.
- 802.1p priority (VLAN priority).
- Link layer protocol type.
- Encapsulation type.

Procedure

1. Enter system view.

   system-view

2. Create a Layer 2 ACL and enter its view. Choose one option as needed:
   - Create a Layer 2 ACL by specifying an ACL number.

     acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]
   - Create a Layer 2 ACL by specifying the **mac** keyword.

     acl mac { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

3. (Optional.) Configure a description for the Layer 2 ACL.

   description *text*

   By default, a Layer 2 ACL does not have a description.

4. (Optional.) Enable rule ID preemption.

   rule insert-only enable

   By default, rule ID preemption is disabled.

5. (Optional.) Set the rule numbering step.

   step *step-value*

   By default, the rule numbering step is 5 and the start rule ID is 0.

6. Create or edit a rule.

   rule [*rule-id*] { **deny** | **permit** } [**cos** *dot1p* | **counting** | **dest-mac** *dest-address dest-mask* | { **lsap** *lsap-type lsap-type-mask* | **type** *protocol-type protocol-type-mask* } | **source-mac** *source-address source-mask* | **time-range** *time-range-name*] *

7. (Optional.) Add or edit a rule comment.

   rule *rule-id* **comment** *text*

   By default, no rule comment is configured.

Copying an ACL

About this task

You can create an ACL by copying an existing ACL (source ACL). The new ACL (destination ACL) has the same properties and content as the source ACL, but uses a different number or name than the source ACL.

Restrictions and guidelines

To successfully copy an ACL, make sure:

- The destination ACL is the same type as the source ACL.
- The source ACL already exists, but the destination ACL does not.

Procedure

1. Enter system view.

   system-view

2. Copy an existing ACL to create a new ACL.

   acl [**ipv6** | **mac**] **copy** { *source-acl-number* | **name** *source-acl-name* } **to** { *dest-acl-number* | **name** *dest-acl-name* }

Enabling ACL acceleration

About this task

ACL acceleration speeds up ACL rule lookup. The acceleration effect increases with the number of ACL rules. For example, when a large ACL is used for a session-based service, such as NAT or ASPF, ACL acceleration can avoid session timeouts caused by ACL processing delays.

ACL acceleration is delayed for a period after an ACL rule is added, deleted, or modified. If additional rule changes occur during the delay period, the delay period starts to count again. If an ACL contains 100 or less rules, the delay period is 2 seconds. If an ACL contains more than 100 rules, the delay period is 20 seconds.

Procedure

1. Enter system view.

   system-view

2. Create an ACL and enter ACL view.

   acl { [**ipv6**] { **advanced** | **basic** } { *acl-number* | **name** *acl-name* } | **mac** { *acl-number* | **name** *acl-name* } } [**match-order** { **auto** | **config** }]

3. Enable ACL acceleration for the ACL.

   `accelerate`

   By default, ACL acceleration is disabled.

> △ **CAUTION:**
>
> When there are a large number of ACLs on the device, executing the **`undo accelerate`** command might cause the CPU usage of the device to reach the upper threshold and cause service processing exceptions.

Configuring packet filtering with ACLs

About packet filtering with ACLs

This section describes procedures for using an ACL to filtering packets. For example, you can apply an ACL to an interface to filter incoming or outgoing packets.

Applying an ACL to an interface for packet filtering

Restrictions and guidelines

You can apply a maximum of 32 ACLs to the same direction of an interface.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Apply an ACL to the interface to filter packets.

   `packet-filter` [**`ipv6`** | **`mac`**] { *`acl-number`* | **`name`** *`acl-name`* } **`inbound`**

   By default, an interface does not filter packets.

Applying an ACL to a zone pair for packet filtering

Restrictions and guidelines

You can apply a maximum of 32 ACLs to the same zone pair. For more information about zone pairs, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter zone pair view.

   `zone-pair security source` *`source-zone-name`* **`destination`** *`destination-zone-name`*

3. Apply an ACL to the zone pair to filter packets.

   `packet-filter` [**`ipv6`**] { *`acl-number`* | **`name`** *`acl-name`* }

   By default, a zone pair does not filter packets.

Configuring logging and SNMP notifications for packet filtering

About this task

You can configure the ACL module to generate log entries or SNMP notifications for packet filtering and output them to the information center or SNMP module at the output interval. The log entry or notification records the number of matching packets and the matched ACL rules. When the first packet of a flow matches an ACL rule, the output interval starts, and the device immediately outputs a log entry or notification for this packet. When the output interval ends, the device outputs a log entry or notification for subsequent matching packets of the flow.

For more information about the information center, see *Network Management and Monitoring Configuration Guide.*

For more information about SNMP, see *Network Management and Monitoring Configuration Guide.*

Procedure

1. Enter system view.

   `system-view`

2. Set the interval for outputting packet filtering logs or notifications.

   `acl` { **`logging`** | **`trap`** } **`interval`** *`interval`*

   The default setting is 0 minutes. By default, the device does not generate log entries or SNMP notifications for packet filtering.

Setting the packet filtering default action

About this task

By default, the packet filter permits packets that do not match any ACL rule to pass. Perform this task to deny packets that do not match any ACL rule. The packet filtering default action does not take effect on zone pair packet filtering. The default action for zone pair packet filtering is always **deny**.

Procedure

1. Enter system view.

   `system-view`

2. Set the packet filtering default action to deny.

   `packet-filter default deny`

   By default, the packet filter permits packets that do not match any ACL rule to pass.

Display and maintenance commands for ACL

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display ACL configuration and match statistics.	**display acl** [**ipv6** | **mac**] { *acl-number* | **all** | **name** *acl-name* }
Display ACL acceleration status.	**display acl accelerate** { **summary** [**ipv6** | **mac**] | **verbose** [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* } **slot** *slot-number* }
Display ACL application information	**display packet-filter** { **interface**

Task	Command
for packet filtering.	[*interface-type interface-number*] [**inbound**] | **zone-pair security** [**source** *source-zone-name* **destination** *destination-zone-name*] } [**slot** *slot-number*]
Display match statistics for packet filtering ACLs.	**display packet-filter statistics** { **interface** *interface-type interface-number* **inbound** [**default** | [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }] | **zone-pair security source** *source-zone-name* **destination** *destination-zone-name* [[**ipv6**] { *acl-number* | **name** *acl-name* }] } [**brief**]
Display the accumulated statistics for packet filtering ACLs.	**display packet-filter statistics sum inbound** [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* } [**brief**]
Display detailed ACL packet filtering information.	**display packet-filter verbose** { **interface** *interface-type interface-number* **inbound** [[**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }] | **zone-pair security source** *source-zone-name* **destination** *destination-zone-name* [[**ipv6**] { *acl-number* | **name** *acl-name* }] } [**slot** *slot-number*]
Clear ACL statistics.	**reset acl** [**ipv6** | **mac**] **counter** { *acl-number* | **all** | **name** *acl-name* }
Clear match statistics for packet filtering ACLs.	**reset packet-filter statistics** { **interface** [*interface-type interface-number*] **inbound** [**default** | [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }] | **zone-pair security** [**source** *source-zone-name* **destination** *destination-zone-name*] [**ipv6**] { *acl-number* | **name** *acl-name* } }

ACL configuration examples

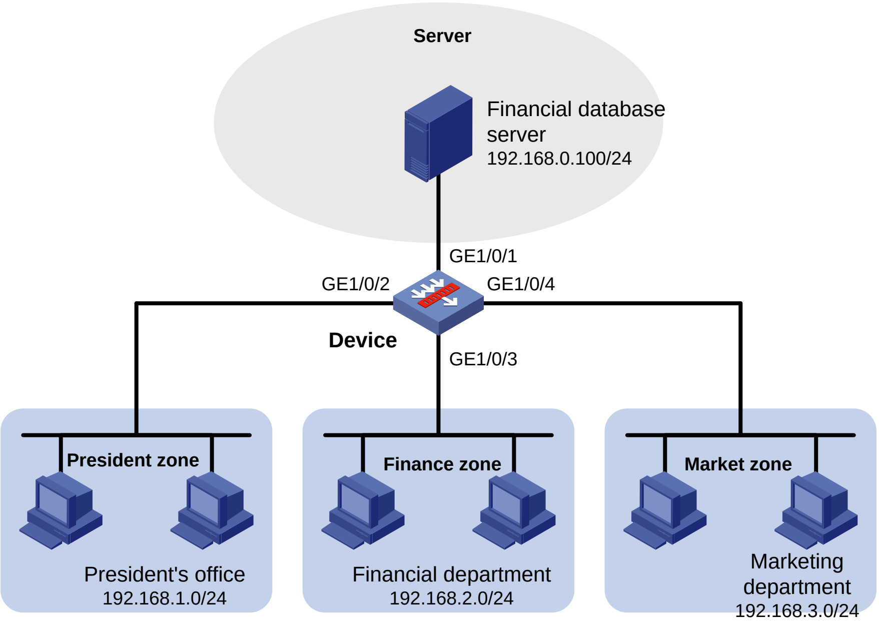
Example: Configuring a zone pair-based packet filter

Network configuration

A company interconnects its departments through the device. The financial database server, President's office, Financial department, and Marketing department belong to different security zones. Configure a packet filter to:

- Permit access from the President's office at any time to the financial database server.
- Permit access from the Financial department to the financial database server only during working hours (from 8:00 to 18:00) on working days.
- Deny access from any other department to the financial database server.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces:

   # Assign an IP address to GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 192.168.0.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Add interfaces to security zones.

   ```
   [Device] security-zone name Server
   [Device-security-zone-Server] import interface gigabitethernet 1/0/1
   [Device-security-zone-Server] quit
   [Device] security-zone name President
   [Device-security-zone-President] import interface gigabitethernet 1/0/2
   [Device-security-zone-President] quit
   [Device] security-zone name Finance
   [Device-security-zone-Finance] import interface gigabitethernet 1/0/3
   [Device-security-zone-Finance] quit
   [Device] security-zone name Market
   [Device-security-zone-Market] import interface gigabitethernet 1/0/4
   [Device-security-zone-Market] quit
   ```

3. Create a periodic time range from 8:00 to 18:00 on working days.

   ```
   [Device] time-range work 08:0 to 18:00 working-day
   ```

4. Configure ACLs:

   # Configure ACL 3000 to permit access from the President's office at any time to the financial database server.

   ```
   [Device] acl advanced 3000
   [Device-acl-ipv4-adv-3000] rule permit ip source 192.168.1.0 0.0.0.255 destination
   192.168.0.100 0
   ```

```
[Device-acl-ipv4-adv-3000] quit
```

# Configure ACL 3001 to permit access from the Financial department to the financial database server only during working hours on working days.

```
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3001] rule permit ip source 192.168.2.0 0.0.0.255 destination
192.168.0.100 0 time-range work
[Device-acl-ipv4-adv-3001] quit
```

# Configure ACL 3002 to deny access from any other department to the financial database server.

```
[Device] acl advanced 3002
[Device-acl-ipv4-adv-3002] rule deny ip source any destination 192.168.0.100 0
[Device-acl-ipv4-adv-3002] quit
```

5. Applying ACLs to zone pairs for packet filtering:

# Create a zone pair with the source security zone **President** and destination security zone **Server**. Apply ACL 3000 to the zone pair for packet filtering.

```
[Device] zone-pair security source president destination server
[Device-zone-pair-security-President-Server] packet-filter 3000
[Device-zone-pair-security-President-Server] quit
```

# Create a zone pair with the source security zone **Finance** and destination security zone **Server**. Apply ACL 3001 to the zone pair for packet filtering.

```
[Device] zone-pair security source finance destination server
[Device-zone-pair-security-Finance-Server] packet-filter 3001
[Device-zone-pair-security-President-Server] quit
```

# Create a zone pair with the source security zone **Market** and destination security zone **Server**. Apply ACL 3002 to the zone pair for packet filtering.

```
[Device] zone-pair security source market destination server
[Device-zone-pair-security-Market-Server] packet-filter 3002
[Device-zone-pair-security-Market-Server] quit
```

Verifying the configuration

# Verify that a PC in the Financial department can ping the database server during working hours. (All PCs in this example use Windows XP).

```
C:\> ping 192.168.0.100


Pinging 192.168.0.100 with 32 bytes of data:


Reply from 192.168.0.100: bytes=32 time=1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

# Verify that a PC in the Marketing department cannot ping the database server during working hours.

```
C:\> ping 192.168.0.100


Pinging 192.168.0.100 with 32 bytes of data:
```

```
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

# Display configuration and match statistics for IPv4 advanced ACL 3001 and 3002 on the device during working hours.

```
[Device] display acl 3001
Advanced IPv4 ACL 3001, 1 rule,
ACL's step is 5
 rule 0 permit ip source 192.168.2.0 0.0.0.255 destination 192.168.0.100 0 time-range work
(4 times matched) (Active)
[Device] display acl 3002
Advanced IPv4 ACL 3002, 1 rule,
ACL's step is 5
 rule 0 deny ip destination 192.168.0.100 0 (4 times matched)
```

The output shows that the rule in ACL 3001 is active. ACL 3001 and ACL 3002 both have been matched four times as the result of the ping operations.

Contents

QoS overview

In data communications, Quality of Service (QoS) provides differentiated service guarantees for diversified traffic in terms of bandwidth, delay, jitter, and drop rate, all of which can affect QoS.

QoS manages network resources and prioritizes traffic to balance system resources.

The following section describes typical QoS service models and widely used QoS techniques.

QoS service models

This section describes several typical QoS service models.

Best-effort service model

The best-effort model is a single-service model. The best-effort model is not as reliable as other models and does not guarantee delay-free delivery.

The best-effort service model is the default model for the Internet and applies to most network applications. It uses the First In First Out (FIFO) queuing mechanism.

IntServ model

The integrated service (IntServ) model is a multiple-service model that can accommodate diverse QoS requirements. This service model provides the most granularly differentiated QoS by identifying and guaranteeing definite QoS for each data flow.

In the IntServ model, an application must request service from the network before it sends data. IntServ signals the service request with the RSVP. All nodes receiving the request reserve resources as requested and maintain state information for the application flow.

The IntServ model demands high storage and processing capabilities because it requires all nodes along the transmission path to maintain resource state information for each flow. This model is suitable for small-sized or edge networks. However, it is not suitable for large-sized networks, for example, the core layer of the Internet, where billions of flows are present.

DiffServ model

The differentiated service (DiffServ) model is a multiple-service model that can meet diverse QoS requirements. It is easy to implement and extend. DiffServ does not signal the network to reserve resources before sending data, as IntServ does.

QoS techniques in a network

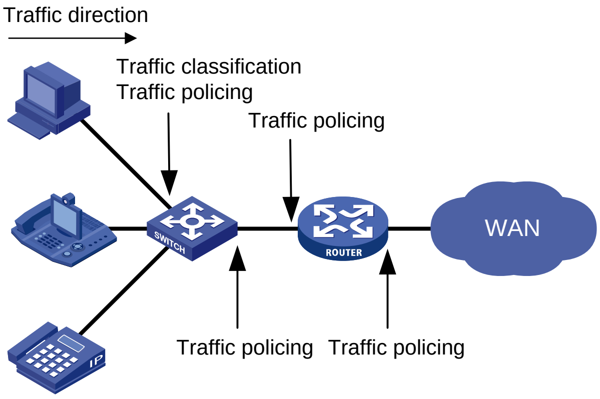
The QoS techniques include the following features:

- Traffic classification.
- Traffic policing.
- Traffic shaping.
- Rate limit.
- Congestion management.
- Congestion avoidance.

The following section briefly introduces these QoS techniques.

All QoS techniques in this document are based on the DiffServ model.

Figure 1 Position of the QoS techniques in a network



As shown in Figure 1, traffic classification, traffic shaping, traffic policing, congestion management, and congestion avoidance mainly implement the following functions:

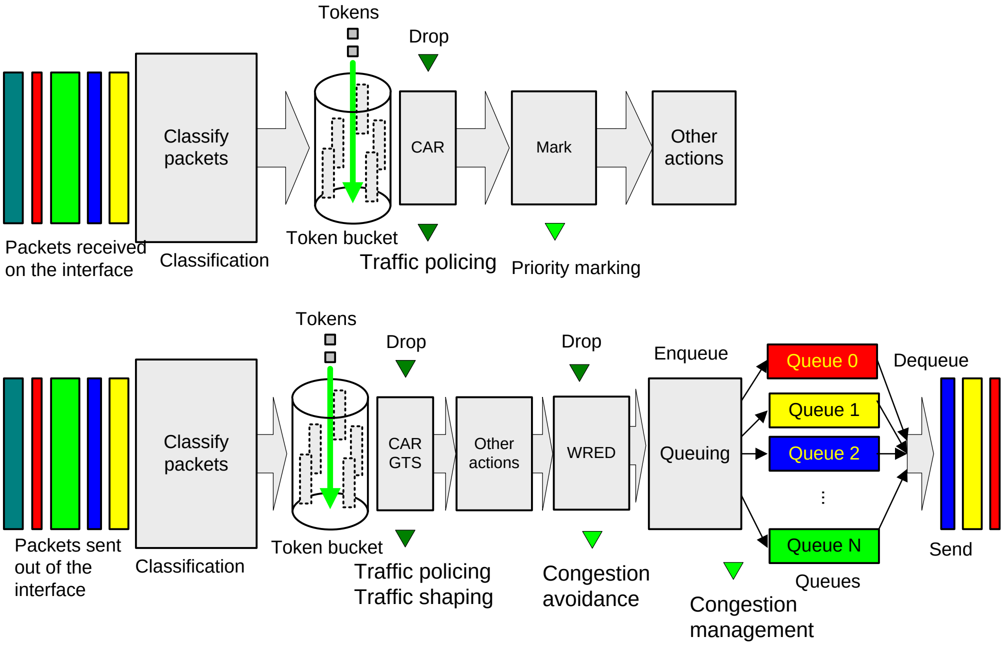
- **Traffic classification**—Uses match criteria to assign packets with the same characteristics to a traffic class. Based on traffic classes, you can provide differentiated services.
- **Traffic policing**—Polices flows and imposes penalties to prevent aggressive use of network resources. You can apply traffic policing to both incoming and outgoing traffic of a port.
- **Traffic shaping**—Adapts the output rate of traffic to the network resources available on the downstream device to eliminate packet drops. Traffic shaping usually applies to the outgoing traffic of a port.
- **Congestion management**—Provides a resource scheduling policy to determine the packet forwarding sequence when congestion occurs. Congestion management usually applies to the outgoing traffic of a port.
- **Congestion avoidance**—Monitors the network resource usage. It is usually applied to the outgoing traffic of a port. When congestion worsens, congestion avoidance reduces the queue length by dropping packets.

QoS processing flow in a device

Figure 2 briefly describes how the QoS module processes traffic.

1. Traffic classifier identifies and classifies traffic for subsequent QoS actions.
2. The QoS module takes various QoS actions on classified traffic as configured, depending on the traffic processing phase and network status. For example, you can configure the QoS module to perform the following operations:
   - Traffic policing for incoming traffic.
   - Traffic shaping for outgoing traffic.
   - Congestion avoidance before congestion occurs.
   - Congestion management when congestion occurs.

Figure 2 QoS processing flow



QoS configuration approaches

You can configure QoS by using the MQC approach or non-MQC approach.

In the modular QoS configuration (MQC) approach, you configure QoS service parameters by using QoS policies. A QoS policy defines QoS actions to take on different classes of traffic and can be applied to an object (such as an interface) to control traffic.

In the non-MQC approach, you configure QoS service parameters without using a QoS policy.

Some features support both approaches, but some support only one.

Configuring a QoS policy

About QoS policies

A QoS policy has the following components:

- **Traffic class**—Defines criteria to match packets.
- **Traffic behavior**—Defines QoS actions to take on matching packets.

By associating a traffic class with a traffic behavior, a QoS policy can perform the QoS actions on matching packets.

A QoS policy can have multiple class-behavior associations.

QoS policy tasks at a glance

To configure a QoS policy, perform the following tasks:

1. Defining a traffic class
2. Defining a traffic behavior
3. Defining a QoS policy
4. (Optional.) Configuring policy nesting
5. Applying the QoS policy
   - Applying the QoS policy to an interface
   - Applying the QoS policy to a control plane
   - Applying the QoS policy in control-plane management view
6. (Optional.) Setting the QoS policy-based traffic rate statistics collection period for an interface

Defining a traffic class

1. Enter system view.
   `system-view`
2. Create a traffic class and enter traffic class view.
   `traffic classifier` *`classifier-name`* [**`operator`** { **`and`** | **`or`** }]
3. Configure a match criterion.
   `if-match` [**`not`**] *`match-criteria`*

   By default, no match criterion is configured.

   For more information, see the **`if-match`** command in *ACL and QoS Command Reference*.

Defining a traffic behavior

1. Enter system view.
   `system-view`
2. Create a traffic behavior and enter traffic behavior view.
   `traffic behavior` *`behavior-name`*
3. Configure an action in the traffic behavior.

By default, no action is configured for a traffic behavior.

For more information about configuring an action, see the subsequent chapters for traffic policing, traffic filtering, priority marking, and so on.

Defining a QoS policy

1. Enter system view.

   system-view

2. Create a QoS policy and enter QoS policy view.

   qos policy *policy-name*

3. Associate a traffic class with a traffic behavior to create a class-behavior association in the QoS policy.

   classifier *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]

   By default, a traffic class is not associated with a traffic behavior.

   Repeat this step to create more class-behavior associations.

Configuring policy nesting

About this task

A QoS policy configuration can contain a parent policy and a child policy.

Policy nesting allows you to create a child policy in the view of a traffic behavior of the parent policy.

You can nest a QoS policy in a traffic behavior to reclassify the traffic class associated with the behavior. Then the system performs the actions defined in the QoS policy on the reclassified traffic. The QoS policy nested in the traffic behavior is called the child policy. The QoS policy that nests the behavior is called the parent policy.

Prerequisites

Before configuring policy nesting, define a child policy (see "Defining a QoS policy").

Procedure

1. Enter system view.

   system-view

2. Define a traffic class for the parent policy.

   a. Create a traffic class for the parent policy and enter traffic class view.

      traffic classifier *classifier-name* [**operator** { **and** | **or** }]

   b. Configure a match criterion.

      if-match [**not**] *match-criteria*

      By default, no match criterion is configured.

      For more information about configuring match criteria, see *ACL and QoS Command Reference*.

   c. Return to system view.

      quit

3. Nest the child QoS policy in the traffic behavior of the parent policy.

   a. Create a traffic behavior for the parent policy and enter traffic behavior view.

      traffic behavior *behavior-name*

   b. Nest the child QoS policy.

      traffic-policy *policy-name*

      By default, policy nesting is not configured.

   c. Return to system view.

      quit

4. Create the parent policy and enter parent policy view.

   qos policy *policy-name*

5. Associate the class with the behavior in the parent policy.

   classifier *classifier-name* **behavior** *behavior-name*

   By default, a class is not associated with a behavior.

Applying the QoS policy

Application destinations

You can apply a QoS policy to the following destinations:

- **Interface**—The QoS policy can be applied to the traffic sent or received on the interface.
- **Control plane**—The QoS policy can be applied to the traffic received on the control plane.
- **Management interface control plane**—The QoS policy can be applied to the traffic sent from the management interface to the control plane.

Restrictions and guidelines for applying a QoS policy

You can modify traffic classes, traffic behaviors, and class-behavior associations in a QoS policy even after it is applied. If a traffic class uses an ACL for traffic classification, you can delete or modify the ACL.

If an action in a traffic behavior cannot take effect, all other actions in the traffic behavior do not take effect.

Applying the QoS policy to an interface

Restrictions and guidelines

A QoS policy can be applied to multiple interfaces. However, only one QoS policy can be applied to one direction (inbound or outbound) of an interface.

The QoS policy applied to the outgoing traffic on an interface does not regulate local packets. Local packets refer to critical protocol packets sent by the local system for operation maintenance. The most common local packets include link maintenance, RIP, and SSH packets.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Apply the QoS policy to the interface.

   qos apply policy *policy-name* { **inbound** | **outbound** }

   By default, no QoS policy is applied to an interface.

Applying the QoS policy to a control plane

About this task

A device provides the user plane and the control plane.

- **User plane**—The units at the user plane are responsible for receiving, transmitting, and switching (forwarding) packets, such as various dedicated forwarding chips. They deliver super processing speeds and throughput.
- **Control plane**—The units at the control plane are processing units running most routing and switching protocols. They are responsible for protocol packet resolution and calculation, such as CPUs. Compared with user plane units, the control plane units allow for great packet processing flexibility but have lower throughput.

When the user plane receives packets that it cannot recognize or process, it transmits them to the control plane. If the transmission rate exceeds the processing capability of the control plane, the control plane will be busy handling undesired packets. As a result, the control plane will fail to handle legitimate packets correctly or timely. As a result, protocol performance is affected.

To address this problem, apply a QoS policy to the control plane to take QoS actions, such as traffic filtering or traffic policing, on inbound traffic. This ensures that the control plane can correctly receive, transmit, and process packets.

A predefined control plane QoS policy uses the protocol type or protocol group type to identify the type of packets sent to the control plane. You can use protocol types or protocol group types in **`if-match`** commands in traffic class view for traffic classification. Then you can reconfigure traffic behaviors for these traffic classes as required. You can use the **`display qos policy control-plane pre-defined`** command to display predefined control plane QoS policies.

Procedure

1. Enter system view.

   `system-view`

2. Enter control plane view.

   `control-plane slot` *`slot-number`*

3. Apply the QoS policy to the control plane.

   `qos apply policy` *`policy-name`* **`inbound`**

   By default, no QoS policy is applied to a control plane.

Applying the QoS policy in control-plane management view

About this task

If the rate of packets from the management interface to the control plane exceeds the processing capability, the control plane will fail to handle the packets correctly or timely. As a result, protocol performance is affected.

This feature allows you to rate limit the packets sent from the management interface to the control plane. This ensures that the control plane can correctly receive, transmit, and process packets from the management interface.

By default, a predefined QoS policy is applied in control-plane management view. To display the predefined QoS policy, use the **`display qos policy control-plane management pre-defined`** command. The predefined QoS policy uses the protocol type or protocol group type to identify the type of packets sent from the management interface to the control plane. You can use protocol types or protocol group types in **`if-match`** commands in traffic class view for traffic classification. Then, you can reconfigure traffic behaviors for these traffic classes as required.

Procedure

1. Enter system view.

   system-view

2. Enter control-plane management view.

   control-plane management

3. Apply the QoS policy.

   qos apply policy *policy-name* **inbound**

   By default, no QoS policy is applied in control-plane management view.

Setting the QoS policy-based traffic rate statistics collection period for an interface

About this task

You can enable collection of per-class traffic statistics over a period of time, including the average forwarding rate and drop rate. For example, if you set the statistics collection period to n minutes, the system performs the following operations:

- Collects traffic statistics for the most recent n minutes.
- Refreshes the statistics every n/5 minutes.

You can use the **display qos policy interface** command to view the collected traffic rate statistics.

A subinterface uses the statistics collection period configured on the main interface.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the traffic rate statistics collection period for the interface.

   qos flow-interval *interval*

   The default setting is 5 minutes.

Display and maintenance commands for QoS policies

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display QoS policy configuration.	**display qos policy user-defined** [*policy-name* [**classifier** *classifier-name*]] [**slot** *slot-number*]
Display QoS policies applied to hub-spoke tunnels on a tunnel interface.	**display qos policy advpn tunnel** *number* [*ipv4-address* | *ipv6-address*] [**outbound**]
Display information about QoS policies	**display qos policy control-plane slot**

Task	Command
applied to the control plane.	*slot-number*
Display information about the predefined QoS policy applied in control-plane management view.	**display qos policy control-plane management pre-defined**
Display information about the predefined QoS policy applied to the control plane.	**display qos policy control-plane pre-defined** [**slot** *slot-number*]
Display information about QoS policies applied to interfaces.	**display qos policy interface** [*interface-type interface-number*] [**slot** *slot-number*] [**inbound** | **outbound**]
Display traffic behavior configuration.	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]
Display traffic class configuration.	**display traffic classifier user-defined** [*classifier-name*] [**slot** *slot-number*]
Clear the statistics for the QoS policy applied to the control plane.	**reset qos policy control-plane slot** *slot-number*
Clear the statistics for QoS policies applied to hub-spoke tunnels on a tunnel interface.	**reset qos policy advpn tunnel** *number* [*ipv4-address* | *ipv6-address*] [**outbound**]

Configuring traffic policing, GTS, and rate limit

About traffic policing, GTS, and rate limit

Traffic limit helps assign network resources (including bandwidth) and increase network performance. For example, you can configure a flow to use only the resources committed to it in a certain time range. This avoids network congestion caused by burst traffic.

Traffic policing, Generic Traffic Shaping (GTS), and rate limit control the traffic rate and resource usage according to traffic specifications. You can use token buckets for evaluating traffic specifications.

Traffic evaluation and token buckets

Token bucket features

A token bucket is analogous to a container that holds a certain number of tokens. Each token represents a certain forwarding capacity. The system puts tokens into the bucket at a constant rate. When the token bucket is full, the extra tokens cause the token bucket to overflow.

Evaluating traffic with the token bucket

A token bucket mechanism evaluates traffic by looking at the number of tokens in the bucket. If the number of tokens in the bucket is enough for forwarding the packets:

- The traffic conforms to the specification (called conforming traffic).
- The corresponding tokens are taken away from the bucket.

Otherwise, the traffic does not conform to the specification (called excess traffic).

A token bucket has the following configurable parameters:

- Mean rate at which tokens are put into the bucket, which is the permitted average rate of traffic. It is usually set to the committed information rate (CIR).
- Burst size or the capacity of the token bucket. It is the maximum traffic size permitted in each burst. It is usually set to the committed burst size (CBS). The set burst size must be greater than the maximum packet size.

Each arriving packet is evaluated.

Complicated evaluation

You can set two token buckets, bucket C and bucket E, to evaluate traffic in a more complicated environment and achieve more policing flexibility. The following are main mechanisms used for complicated evaluation:

- **Single rate two color**—Uses one token bucket and the following parameters:
    - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
    - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.

  When a packet arrives, the following rules apply:

    - If bucket C has enough tokens to forward the packet, the packet is colored green.
    - Otherwise, the packet is colored red.
- **Single rate three color**—Uses two token buckets and the following parameters:

    - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
    - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.
    - **EBS**—Size of bucket E minus size of bucket C, which specifies the transient burst of traffic that bucket E can forward. The EBS cannot be 0. The size of E bucket is the sum of the CBS and EBS.

    When a packet arrives, the following rules apply:

    - If bucket C has enough tokens, the packet is colored green.
    - If bucket C does not have enough tokens but bucket E has enough tokens, the packet is colored yellow.
    - If neither bucket C nor bucket E has sufficient tokens, the packet is colored red.
- **Two rate three color**—Uses two token buckets and the following parameters:
    - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
    - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.
    - **PIR**—Rate at which tokens are put into bucket E, which specifies the average packet transmission or forwarding rate allowed by bucket E.
    - **EBS**—Size of bucket E, which specifies the transient burst of traffic that bucket E can forward.

    When a packet arrives, the following rules apply:

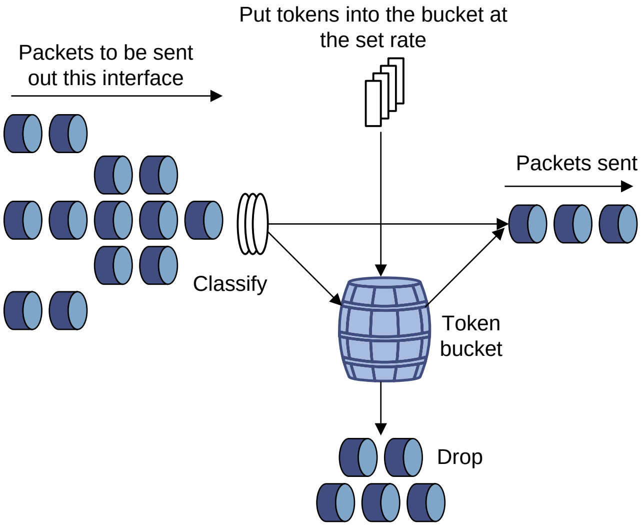
    - If bucket C has enough tokens, the packet is colored green.
    - If bucket C does not have enough tokens but bucket E has enough tokens, the packet is colored yellow.
    - If neither bucket C nor bucket E has sufficient tokens, the packet is colored red.

Traffic policing

Traffic policing supports policing the inbound traffic and the outbound traffic.

A typical application of traffic policing is to supervise the specification of traffic entering a network and limit it within a reasonable range. Another application is to "discipline" the extra traffic to prevent aggressive use of network resources by an application. For example, you can limit bandwidth for HTTP packets to less than 50% of the total. If the traffic of a session exceeds the limit, traffic policing can drop the packets or set the precedence of the packets. Figure 3 shows an example of policing outbound traffic on an interface.

Figure 3 Traffic policing


Put tokens into the bucket at the set rate
Packets to be sent out this interface
Packets sent
Classify
Token bucket
Drop

Traffic policing is widely used in policing traffic entering the ISP networks. It can classify the policed traffic and take predefined policing actions on each packet depending on the evaluation result:

- Forwarding the packet if the evaluation result is "conforming."
- Dropping the packet if the evaluation result is "excess."
- Forwarding the packet with its precedence re-marked if the evaluation result is "conforming."
- Delivering the packet to next-level traffic policing with its precedence re-marked if the evaluation result is "conforming."
- Entering the next-level policing (you can set multiple traffic policing levels, each focused on objects at different levels).

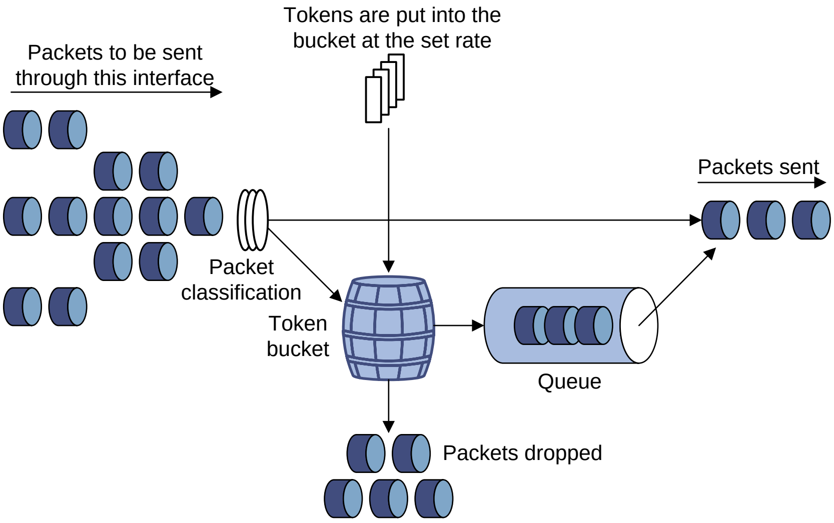
GTS

GTS supports shaping only outbound traffic. GTS limits the outbound traffic rate by buffering exceeding traffic. You can use GTS to adapt the traffic output rate on a device to the input traffic rate of its connected device to avoid packet loss.

The differences between traffic policing and GTS are as follows:

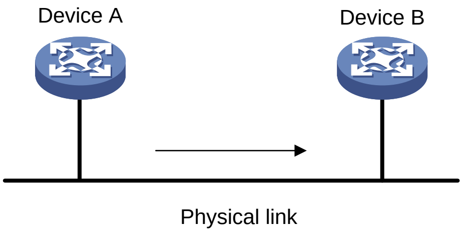
- Packets to be dropped with traffic policing are retained in a buffer or queue with GTS, as shown in Figure 4. When enough tokens are in the token bucket, the buffered packets are sent at an even rate.
- GTS can result in additional delay and traffic policing does not.

Figure 4 GTS


Tokens are put into the bucket at the set rate
Packets to be sent through this interface
Packets sent
Packet classification
Token bucket
Queue
Packets dropped

For example, in Figure 5, Device B performs traffic policing on packets from Device A and drops packets exceeding the limit. To avoid packet loss, you can perform GTS on the outgoing interface of Device A so that packets exceeding the limit are cached in Device A. Once resources are released, GTS takes out the cached packets and sends them out.

Figure 5 GTS application



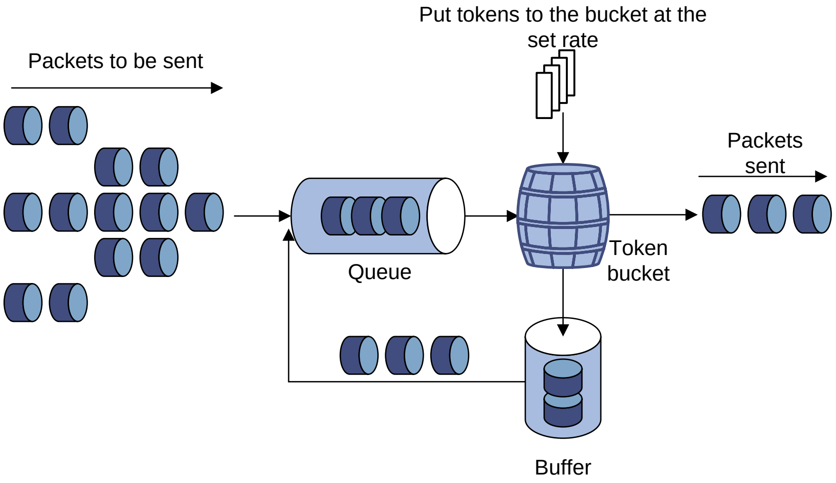
Rate limit

Rate limit controls the rates of inbound and outbound traffic.

The rate limit of an interface or PW specifies the maximum rate for forwarding packets (excluding critical packets).

Rate limit also uses token buckets for traffic control. When rate limit is configured on an interface, a token bucket handles all packets to be sent through the interface for rate limiting. If enough tokens are in the token bucket, packets can be forwarded. Otherwise, packets are put into QoS queues for congestion management. In this way, the traffic passing the interface is controlled.

Figure 6 Rate limit implementation



The token bucket mechanism limits traffic rate when accommodating bursts. It allows bursty traffic to be transmitted if enough tokens are available. If tokens are scarce, packets cannot be transmitted until efficient tokens are generated in the token bucket. It restricts the traffic rate to the rate for generating tokens.

Rate limit controls the total rate of all packets on an interface. It is easier to use than traffic policing in controlling the total traffic rate.

Configuring traffic policing

Traffic policing configuration approaches

You can configure traffic policing by using the MQC approach or the non-MQC approach. If both approaches are used, the MQC configuration takes effect.

You can configure the following types of traffic policing by using the non-MQC approach:

- CAR-list-based traffic policing.
- ACL-based traffic policing.
- Traffic policing for all traffic.

If traffic policing is configured by using both the MQC approach and non-MQC approach, the configuration in MQC approach takes effect.

Configuring traffic policing by using the MQC approach

Restrictions and guidelines

The device supports the following application destinations for traffic policing:

- Interface.
- Control plane.
- Control-plane management view.

Procedure

1. Enter system view.

```
system-view
```

2. Define a traffic class.
   a. Create a traffic class and enter traffic class view.

      traffic classifier *classifier-name* [**operator** { **and** | **or** }]
   b. Configure a match criterion.

      if-match [**not**] *match-criteria*

      By default, no match criterion is configured.

      For more information about the **if-match** command, see *ACL and QoS Command Reference*.
   c. Return to system view.

      quit
3. Define a traffic behavior.
   a. Create a traffic behavior and enter traffic behavior view.

      traffic behavior *behavior-name*
   b. Configure a traffic policing action.
   - In absolute value:

     car cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

     car cir *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *
   - In percentage:

     car cir percent *cir-percent* [**cbs** *cbs-time* [**ebs** *ebs-time*]] [**green** *action* | **red** *action* | **yellow** *action*] *

     car cir percent *cir-percent* [**cbs** *cbs-time*] **pir percent** *pir-percent* [**ebs** *ebs-time*] [**green** *action* | **red** *action* | **yellow** *action*] *

     By default, no traffic policing action is configured.

     Support for this command depends on the device model. For more information, see the command reference.
   c. Return to system view.

      quit
4. Define a QoS policy.
   a. Create a QoS policy and enter QoS policy view.

      qos policy *policy-name*
   b. Associate the traffic class with the traffic behavior in the QoS policy.

      classifier *classifier-name* **behavior** *behavior-name*

      By default, a traffic class is not associated with a traffic behavior.
   c. Return to system view.

      quit
5. Apply the QoS policy.

   For more information, see "Applying the QoS policy."

   By default, no QoS policy is applied.

Configuring CAR-list-based static traffic policing

1. Enter system view.

   system-view

2. Configure a CAR list.

   qos carl *carl-index* { **dscp** *dscp-list* | **mac** *mac-address* | **mpls-exp** *mpls-exp-value* | **precedence** *precedence-value* | { **destination-ip-address** | **source-ip-address** } { **range** *start-ip-address* **to** *end-ip-address* | **subnet** *ip-address mask-length* } [**per-address** [**shared-bandwidth**]] }

3. Enter interface view.

   interface *interface-type interface-number*

4. Apply a CAR-list-based CAR policy to the interface.

   qos car { **inbound** | **outbound** } **carl** *carl-index* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

   qos car { **inbound** | **outbound** } **carl** *carl-index* **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

   By default, no CAR policy is applied to an interface.

Configuring ACL-based traffic policing

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an ACL-based CAR policy on the interface.

   qos car { **inbound** | **outbound** } **acl** [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

   qos car { **inbound** | **outbound** } **acl** [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

   By default, no CAR policy is configured on an interface.

Configuring traffic policing for all traffic

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a CAR policy for all traffic on the interface.

   qos car { **inbound** | **outbound** } **any** **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

   qos car { **inbound** | **outbound** } **any** **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

   By default, no CAR policy is configured on an interface.

Configuring GTS

GTS configuration approaches

You can configure GTS by using either the MQC approach or non-MQC approach.

You can configure the following types of GTS by using the non-MQC approach:

- ACL-based GTS.
- GTS for all traffic.

If GTS is configured by using both the MQC approach and non-MQC approach, the configuration in MQC approach takes effect.

Hardware compatibility with GTS

Models	**Feature compatibility**
NFNX5-HD6480, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

Configuring GTS by using the MQC approach

Restrictions and guidelines

The device supports the following application destinations for GTS:

- **Interface.**
- **Control plane**.

Procedure

1. Enter system view.
   system-view
2. Define a traffic class.
   a. Create a traffic class and enter traffic class view.
      traffic classifier *classifier-name* [**operator** { **and** | **or** }]
   b. Configure a match criterion.
      if-match [**not**] *match-criteria*
      By default, no match criterion is configured.
      For configurable match criteria, see the **if-match** command in *ACL and QoS Command Reference*.
   c. Return to system view.
      quit
3. Define a traffic behavior.
   a. Create a traffic behavior and enter traffic behavior view.
      traffic behavior *behavior-name*
   b. Configure a GTS action.

- In absolute value:

  gts cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

- In percentage:

  gts percent cir *cir-percent* [**cbs** *cbs-time* [**ebs** *ebs-time*]] [**queue-length** *queue-length*]

By default, no GTS action is configured.

c. Return to system view.

quit

4. Define a QoS policy.

a. Create a QoS policy and enter QoS policy view.

qos policy *policy-name*

b. Associate the class with the traffic behavior in the QoS policy.

classifier *classifier-name* **behavior** *behavior-name*

By default, a traffic class is not associated with a traffic behavior.

c. Return to system view.

quit

5. Apply the QoS policy.

For more information, see "Applying the QoS policy."

By default, no QoS policy is applied.

Configuring ACL-based GTS

1. Enter system view.

system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Configure ACL-based GTS on the interface.

qos gts acl [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

By default, GTS is not configured on an interface.

Configuring GTS for all traffic

1. Enter system view.

system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Configure GTS on the interface.

qos gts any cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

By default, GTS is not configured on an interface.

Configuring the rate limit

Hardware compatibility with rate limit

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	No

Configuring the rate limit for an interface

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure the rate limit for the interface.

   `qos lr outbound cir` *`committed-information-rate`* [**`cbs`** *`committed-burst-size`* [**`ebs`** *`excess-burst-size`*]]

   By default, no rate limit is configured on an interface.

Including the physical layer header in calculating the packet length for rate limiting

About this task

By default, the device calculates the packet length for rate limiting based on the data link layer frame. This feature allows the device to include a 24-byte physical layer header in calculating the packet length for rate limiting.

Restrictions and guidelines

This feature takes effect only on Layer 3 Ethernet interfaces and Layer 3 aggregate interfaces.

Procedure

1. Enter system view.

   `system-view`

2. Include the physical layer header in calculating the packet length for rate limiting.

   `qos overhead layer physical`

   By default, the device calculates the packet length for rate limiting based on the data link layer frame.

Display and maintenance commands for traffic policing, GTS, and rate limit

Execute **`display`** commands in any view.

Task	Command
Display CAR configuration and statistics on an interface.	**display qos car interface** [*interface-type interface-number*]
Display CAR list information.	**display qos carl** [*carl-index*] [**slot** *slot-number*]
Display GTS configuration and statistics for interfaces.	**display qos gts interface** [*interface-type interface-number*]
Display rate limit configuration and statistics.	**display qos lrinterface** [*interface-type interface-number*]
Display traffic behavior configuration.	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]

Configuring traffic filtering

About traffic filtering

You can filter in or filter out traffic of a class by associating the class with a traffic filtering action. For example, you can filter packets sourced from an IP address according to network status.

Restrictions and guidelines: Traffic filtering configuration

The device supports applying traffic filtering to an interface or control plane.

Procedure

1. Enter system view.

   `system-view`

2. Define a traffic class.

   a. Create a traffic class and enter traffic class view.

      traffic classifier *classifier-name* [**operator** { **and** | **or** }]

   b. Configure a match criterion.

      if-match [**not**] *match-criteria*

      By default, no match criterion is configured.

      For more information about configuring match criteria, see *ACL and QoS Command Reference*.

   c. Return to system view.

      `quit`

3. Define a traffic behavior.

   a. Create a traffic behavior and enter traffic behavior view.

      traffic behavior *behavior-name*

   b. Configure the traffic filtering action.

      filter { **deny** | **permit** }

      By default, no traffic filtering action is configured.

      If a traffic behavior has the **filter deny** action, all other actions in the traffic behavior do not take effect.

   c. Return to system view.

      `quit`

4. Define a QoS policy.

   a. Create a QoS policy and enter QoS policy view.

      qos policy *policy-name*

   b. Associate the traffic class with the traffic behavior in the QoS policy.

      classifier *classifier-name* **behavior** *behavior-name*

      By default, a traffic class is not associated with a traffic behavior.

    c. Return to system view.

       quit

5. Apply the QoS policy.

   For more information, see "Applying the QoS policy."

   By default, no QoS policy is applied.

6. (Optional.) Display the traffic filtering configuration.

   display traffic behavior user-defined [*behavior-name*] [**slot** *slot-number*]

   This command is available in any view.

Configuring priority marking

About priority marking

Priority marking sets the priority fields or flag bits of packets to modify the priority of packets. For example, you can use priority marking to set IP precedence or DSCP for a class of IP packets to control the forwarding of these packets.

To configure priority marking to set the priority fields or flag bits for a class of packets, perform the following tasks:

1. Configure a traffic behavior with a priority marking action.
2. Associate the traffic class with the traffic behavior.

Configuring priority marking

Restrictions and guidelines

The device supports applying priority marking to an interface or control plane.

Procedure

1. Enter system view.

   system-view

2. Define a traffic class.

   a. Create a traffic class and enter traffic class view.

      traffic classifier *classifier-name* [**operator** { **and** | **or** }]

   b. Configure a match criterion.

      if-match [**not**] *match-criteria*

      By default, no match criterion is configured.

      For more information about the **if-match** command, see *ACL and QoS Command Reference*.

   c. Return to system view.

      quit

3. Define a traffic behavior.

   a. Create a traffic behavior and enter traffic behavior view.

      traffic behavior *behavior-name*

   b. Configure a priority marking action.

      For configurable priority marking actions, see the **remark** commands in *ACL and QoS Command Reference*.

   c. Return to system view.

      quit

4. Define a QoS policy.

   a. Create a QoS policy and enter QoS policy view.

      qos policy *policy-name*

   b. Associate the traffic class with the traffic behavior in the QoS policy.

      classifier *classifier-name* **behavior** *behavior-name*

      By default, a traffic class is not associated with a traffic behavior.

c. Return to system view.

   quit

5. Apply the QoS policy.

   For more information, see "Applying the QoS policy."

   By default, no QoS policy is applied.

6. (Optional.) Display the priority marking configuration.

   display traffic behavior user-defined [*behavior-name*] [**slot** *slot-number*]

   This command is available in any view.

Appendixes

Appendix A Acronyms

Table 1 Appendix A Acronyms

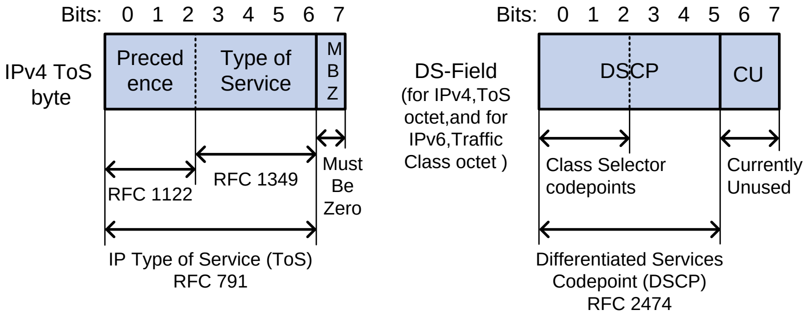
Acronym	Full spelling
AF	Assured Forwarding
BE	Best Effort
BQ	Bandwidth Queuing
CAR	Committed Access Rate
CBS	Committed Burst Size
CBQ	Class Based Queuing
CE	Congestion Experienced
CIR	Committed Information Rate
CQ	Custom Queuing
DiffServ	Differentiated Service
DSCP	Differentiated Services Code Point
EBS	Excess Burst Size
ECN	Explicit Congestion Notification
EF	Expedited Forwarding
FIFO	First in First out
FQ	Fair Queuing
GMB	Guaranteed Minimum Bandwidth
GTS	Generic Traffic Shaping
IntServ	Integrated Service
ISP	Internet Service Provider
LLQ	Low Latency Queuing
LSP	Label Switched Path
MPLS	Multiprotocol Label Switching
PE	Provider Edge
PIR	Peak Information Rate
PQ	Priority Queuing
PW	Pseudowire
QoS	Quality of Service
QPPB	QoS Policy Propagation Through the Border Gateway Protocol
RED	Random Early Detection
RSVP	Resource Reservation Protocol

Acronym	Full spelling
RTP	Real-Time Transport Protocol
SP	Strict Priority
ToS	Type of Service
VPN	Virtual Private Network
WFQ	Weighted Fair Queuing
WRED	Weighted Random Early Detection
WRR	Weighted Round Robin

Appendix B Introduction to packet precedence

IP precedence and DSCP values

Figure 7 ToS and DS fields



Bits: 0 1 2 3 4 5 6 7
IPv4 ToS byte
Preced ence
Type of Service
M B Z
RFC 1122
RFC 1349
Must Be Zero
IP Type of Service (ToS) RFC 791
Bits: 0 1 2 3 4 5 6 7
DS-Field (for IPv4,ToS octet,and for IPv6,Traffic Class octet)
DSCP
CU
Class Selector codepoints
Currently Unused
Differentiated Services Codepoint (DSCP) RFC 2474

As shown in Figure 7, the ToS field in the IP header contains 8 bits. The first 3 bits (0 to 2) represent IP precedence from 0 to 7. According to RFC 2474, the ToS field is redefined as the differentiated services (DS) field. A DSCP value is represented by the first 6 bits (0 to 5) of the DS field and is in the range 0 to 63. The remaining 2 bits (6 and 7) are reserved.

Table 2 IP precedence

IP precedence (decimal)	IP precedence (binary)	Description
0	000	Routine
1	001	priority
2	010	immediate
3	011	flash
4	100	flash-override
5	101	critical
6	110	internet
7	111	network

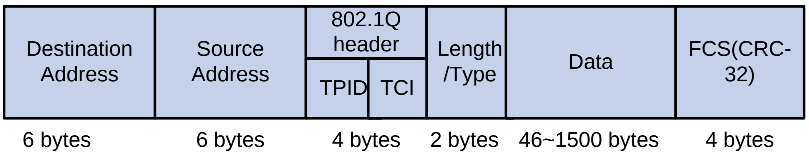
Table 3 DSCP values

DSCP value (decimal)	DSCP value (binary)	Description
46	101110	ef
10	001010	af11
12	001100	af12
14	001110	af13
18	010010	af21
20	010100	af22
22	010110	af23
26	011010	af31
28	011100	af32
30	011110	af33
34	100010	af41
36	100100	af42
38	100110	af43
8	001000	cs1
16	010000	cs2
24	011000	cs3
32	100000	cs4
40	101000	cs5
48	110000	cs6
56	111000	cs7
0	000000	be (default)

802.1p priority

802.1p priority lies in the Layer 2 header. It applies to occasions where Layer 3 header analysis is not needed and QoS must be assured at Layer 2.

Figure 8 An Ethernet frame with an 802.1Q tag header



As shown in Figure 8, the 4-byte 802.1Q tag header contains the 2-byte tag protocol identifier (TPID) and the 2-byte tag control information (TCI). The value of the TPID is 0x8100. Figure 9 shows the format of the 802.1Q tag header. The Priority field in the 802.1Q tag header is called 802.1p priority, because its use is defined in IEEE 802.1p. Table 4 shows the values for 802.1p priority.

Figure 9 802.1Q tag header

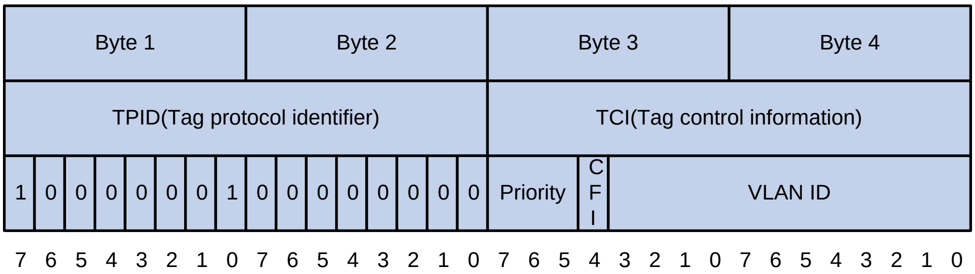


Table 4 Description on 802.1p priority

802.1p priority (decimal)	802.1p priority (binary)	Description
0	000	best-effort
1	001	background
2	010	spare
3	011	excellent-effort
4	100	controlled-load
5	101	video
6	110	voice
7	111	network-management

Contents

Configuring time ranges

About time ranges

You can implement a service based on the time of the day by applying a time range to it. A time-based service takes effect only in time periods specified by the time range. For example, you can implement time-based ACL rules by applying a time range to them.

The following basic types of time ranges are available:

- **Periodic time range**—Recurs periodically on a day or days of the week.
- **Absolute time range**—Represents only a period of time and does not recur.

The active period of a time range is calculated as follows:

1. Combining all periodic statements.
2. Combining all absolute statements.
3. Taking the intersection of the two statement sets as the active period of the time range.

Restrictions and guidelines: Time range configuration

When you configure the ACL hardware mode, follow these restrictions and guidelines:

- If a time range does not exist, the service based on the time range does not take effect.
- You can create a maximum of 1024 time ranges, each with a maximum of 32 periodic statements and 12 absolute statements.

Procedure

1. Enter system view.

   `system-view`

2. Create or edit a time range.

   `time-range` *`time-range-name`* { *`start-time`* **`to`** *`end-time`* *`days`* [**`from`** *`time1 date1`*] [**`to`** *`time2 date2`*] | **`from`** *`time1 date1`* [**`to`** *`time2 date2`*] | **`to`** *`time2 date2`* }

   If an existing time range name is provided, this command adds a statement to the time range.

Display and maintenance commands for time ranges

Execute the **`display`** command in any view.

Task	Command

Display time range configuration and status.	**`display time-range`**
	{ *`time-range-name`* | **`all`** }

NSFOCUS Firewall Series

NF IP Multicast Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for IP multicast features, including multicast overview, multicast routing and forwarding, PIM, IPv6 multicast routing, MLD and forwardingand IGMP.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

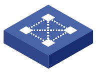
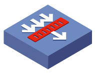
GUI conventions

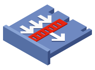
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Multicast overview

Introduction to multicast

As a technique that coexists with unicast and broadcast, the multicast technique effectively addresses the issue of point-to-multipoint data transmission. By enabling high-efficiency point-to-multipoint data transmission over a network, multicast greatly saves network bandwidth and reduces network load.

By using multicast technology, a network operator can easily provide bandwidth-critical and time-critical information services. These services include live webcasting, Web TV, distance learning, telemedicine, Web radio, and real-time video conferencing.

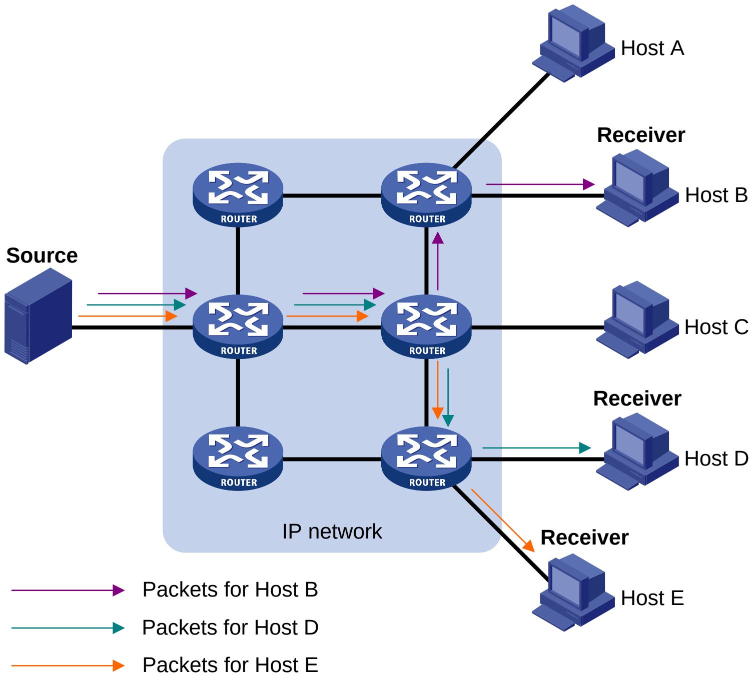
Information transmission techniques

The information transmission techniques include unicast, broadcast, and multicast.

Unicast

In unicast transmission, the information source must send a separate copy of information to each host that needs the information.

Figure 1 Unicast transmission



In Figure 1, Host B, Host D, and Host E need the information. A separate transmission channel must be established from the information source to each of these hosts.

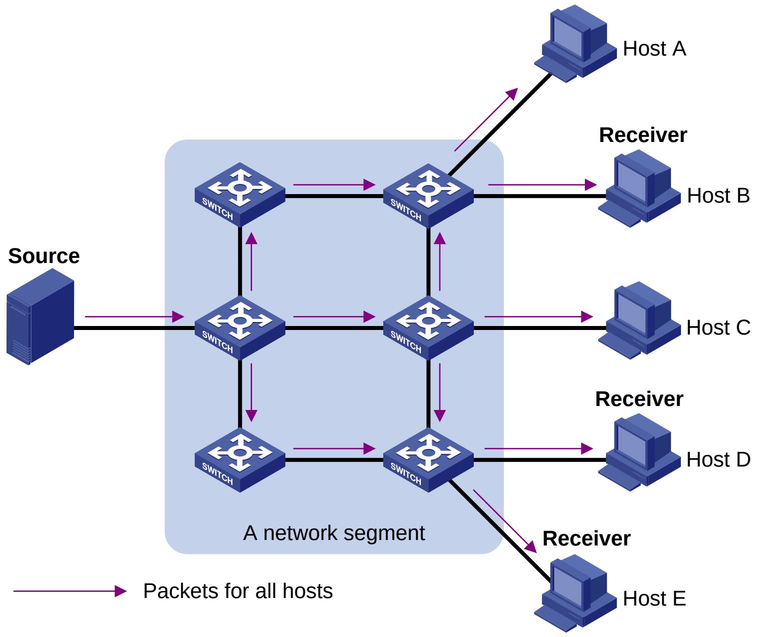
In unicast transmission, the traffic transmitted over the network is proportional to the number of hosts that need the information. If a large number of hosts need the information, the information source must send a separate copy of the same information to each of these hosts. Sending many copies can place a tremendous pressure on the information source and the network bandwidth.

Unicast is not suitable for batch transmission of information.

Broadcast

In broadcast transmission, the information source sends information to all hosts on the subnet, even if some hosts do not need the information.

Figure 2 Broadcast transmission



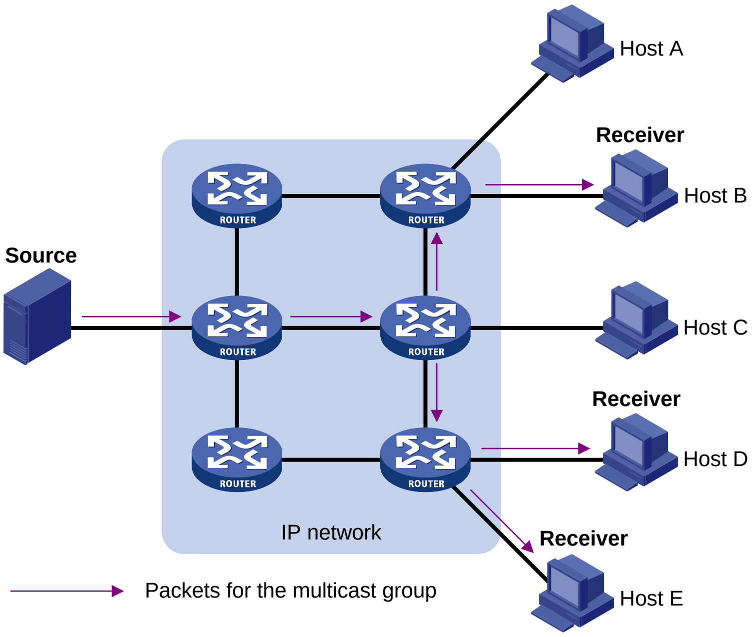
In Figure 2, only Host B, Host D, and Host E need the information. If the information is broadcast to the subnet, Host A and Host C also receive it. In addition to information security issues, broadcasting to hosts that do not need the information also causes traffic flooding on the same subnet.

Broadcast is disadvantageous in transmitting data to specific hosts. Moreover, broadcast transmission is a significant waste of network resources.

Multicast

Multicast provides point-to-multipoint data transmissions with the minimum network consumption. When some hosts on the network need multicast information, the information sender, or multicast source, sends only one copy of the information. Multicast distribution trees are built through multicast routing protocols, and the packets are replicated only on nodes where the trees branch.

Figure 3 Multicast transmission



In Figure 3, the multicast source sends only one copy of the information to a multicast group. Host B, Host D, and Host E, which are information receivers, must join the multicast group. The routers on the network duplicate and forward the information based on the distribution of the group members. Finally, the information is correctly delivered to Host B, Host D, and Host E.

To summarize, multicast has the following advantages:

- **Advantages over unicast**—Multicast data is replicated and distributed until it flows to the farthest-possible node from the source. The increase of receiver hosts will not remarkably increase the load of the source or the usage of network resources.
- **Advantages over broadcast**—Multicast data is sent only to the receivers that need it. This saves network bandwidth and enhances network security. In addition, multicast data is not confined to the same subnet.

Multicast features

- A multicast group is a multicast receiver set identified by an IP multicast address. Hosts must join a multicast group to become members of the multicast group before they receive the multicast data addressed to that multicast group. Typically, a multicast source does not need to join a multicast group.
- A multicast source is an information sender. It can send data to multiple multicast groups at the same time. Multiple multicast sources can send data to the same multicast group at the same time.
- The group memberships are dynamic. Hosts can join or leave multicast groups at any time. Multicast groups are not subject to geographic restrictions.
- Multicast routers or Layer 3 multicast devices are routers or Layer 3 switches that support Layer 3 multicast. They provide multicast routing and manage multicast group memberships on stub subnets with attached group members. A multicast router itself can be a multicast group member.

For a better understanding of the multicast concept, you can compare multicast transmission to the transmission of TV programs.

Table 1 Comparing TV program transmission and multicast transmission

TV program transmission	Multicast transmission
A TV station transmits a TV program through a channel.	A multicast source sends multicast data to a multicast group.
A user tunes the TV set to the channel.	A receiver joins the multicast group.
The user starts to watch the TV program transmitted by the TV station on the channel.	The receiver starts to receive the multicast data sent by the source to the multicast group.
The user turns off the TV set or tunes to another channel.	The receiver leaves the multicast group or joins another group.

Multicast benefits and applications

Multicast benefits

- **Enhanced efficiency**—Reduces the processor load of information source servers and network devices.
- **Optimal performance**—Reduces redundant traffic.
- **Distributed application**—Enables point-to-multipoint applications at the price of minimum network resources.

Multicast applications

- Multimedia and streaming applications, such as Web TV, Web radio, and real-time video/audio conferencing
- Communication for training and cooperative operations, such as distance learning and telemedicine
- Data warehouse and financial applications (stock quotes)
- Any other point-to-multipoint application for data distribution

Multicast models

Based on how the receivers treat the multicast sources, the multicast models include any-source multicast (ASM), source-filtered multicast (SFM), and source-specific multicast (SSM).

ASM model

In the ASM model, any multicast sources can send information to a multicast group. Receivers can join a multicast group and get multicast information addressed to that multicast group from any multicast sources. In this model, receivers do not know the positions of the multicast sources in advance.

SFM model

The SFM model is derived from the ASM model. To a multicast source, the two models appear to have the same multicast membership architecture.

The SFM model functionally extends the ASM model. The upper-layer software checks the source address of received multicast packets and permits or denies multicast traffic from specific sources.

The receivers obtain the multicast data from only part of the multicast sources. To a receiver, multicast sources are not all valid, but are filtered.

SSM model

The SSM model provides a transmission service that enables multicast receivers to specify the multicast sources in which they are interested.

In the SSM model, receivers have already determined the locations of the multicast sources. This is the main difference between the SSM model and the ASM model. In addition, the SSM model uses a different multicast address range than the ASM/SFM model. Dedicated multicast forwarding paths are established between receivers and the specified multicast sources.

Multicast addresses

IP multicast addresses

IPv4 multicast addresses

IANA assigned the Class D address block (224.0.0.0 to 239.255.255.255) to IPv4 multicast.

Table 2 Class D IP address blocks and description

Address block	Description
224.0.0.0 to 224.0.0.255	Reserved permanent group addresses. The IP address 224.0.0.0 is reserved. Other IP addresses can be used by routing protocols and for topology searching, protocol maintenance, and so on. Table 3 lists common permanent group addresses. A packet destined for an address in this block will not be forwarded beyond the local subnet regardless of the TTL value in the IP header.
224.0.1.0 to 238.255.255.255	Globally scoped group addresses. This block includes the following types of designated group addresses: • **232.0.0.0/8**—SSM group addresses. • **233.0.0.0/8**—Glop group addresses.
239.0.0.0 to 239.255.255.255	Administratively scoped multicast addresses. These addresses are considered locally unique rather than globally unique. You can reuse them in domains administered by different organizations without causing conflicts. For more information, see RFC 2365.

NOTE:

Glop is a mechanism for assigning multicast addresses between different ASs. By filling an AS number into the middle two bytes of 233.0.0.0, you get 255 multicast addresses for that AS. For more information, see RFC 2770.

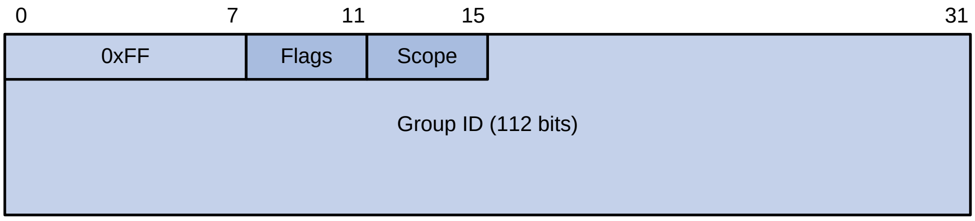
Table 3 Common permanent multicast group addresses

Address	Description
224.0.0.1	All systems on this subnet, including hosts and routers.
224.0.0.2	All multicast routers on this subnet.
224.0.0.3	Unassigned.
224.0.0.4	DVMRP routers.
224.0.0.5	OSPF routers.

Address	Description
224.0.0.6	OSPF designated routers and backup designated routers.
224.0.0.7	Shared Tree (ST) routers.
224.0.0.8	ST hosts.
224.0.0.9	RIPv2 routers.
224.0.0.11	Mobile agents.
224.0.0.12	DHCP server/relay agent.
224.0.0.13	All Protocol Independent Multicast (PIM) routers.
224.0.0.14	RSVP encapsulation.
224.0.0.15	All Core-Based Tree (CBT) routers.
224.0.0.16	Designated SBM.
224.0.0.17	All SBMs.
224.0.0.18	VRRP.

IPv6 multicast addresses

Figure 4 IPv6 multicast format



The following describes the fields of an IPv6 multicast address:

- **0xFF**—The most significant eight bits are 11111111.
- **Flags**—The Flags field contains four bits.

  Figure 5 Flags field format

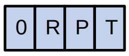
  

  Table 4 Flags field description

Bit	Description
0	Reserved, set to 0.
R	• When set to 0, this address is an IPv6 multicast address without an embedded RP address. • When set to 1, this address is an IPv6 multicast address with an embedded RP address. (The P and T bits must also be set to 1.)
P	• When set to 0, this address is an IPv6 multicast address not based on a unicast prefix. • When set to 1, this address is an IPv6 multicast address based on a unicast prefix. (The T bit must also be set to 1.)

Bit	Description
T	• When set to 0, this address is an IPv6 multicast address permanently-assigned by IANA. • When set to 1, this address is a transient or dynamically assigned IPv6 multicast address.

- **Scope**—The Scope field contains four bits, which represent the scope of the IPv6 internetwork for which the multicast traffic is intended.

Table 5 Values of the Scope field

Value	Meaning
0, F	Reserved.
1	Interface-local scope.
2	Link-local scope.
3	Subnet-local scope.
4	Admin-local scope.
5	Site-local scope.
6, 7, 9 through D	Unassigned.
8	Organization-local scope.
E	Global scope.

- **Group ID**—The Group ID field contains 112 bits. It uniquely identifies an IPv6 multicast group in the scope that the Scope field defines.

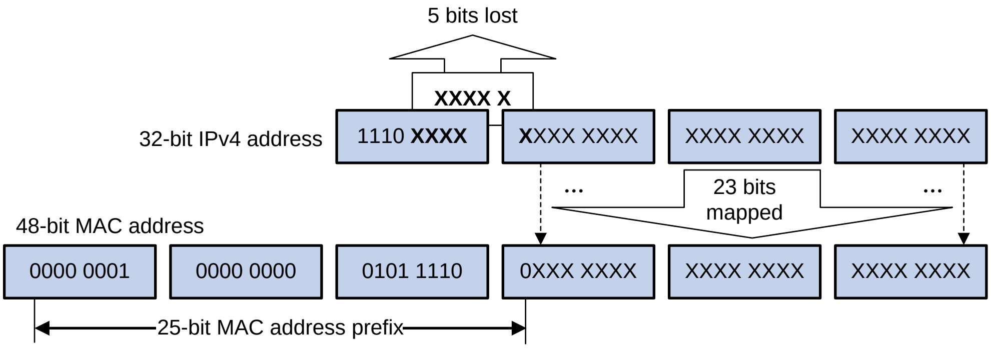
Ethernet multicast MAC addresses

An Ethernet multicast MAC address identifies receivers that belong to the same multicast group at the data link layer.

IPv4 multicast MAC addresses

As defined by IANA, the most significant 24 bits of an IPv4 multicast MAC address are 0x01005E. Bit 25 is 0, and the other 23 bits are the least significant 23 bits of an IPv4 multicast address.

Figure 6 IPv4-to-MAC address mapping

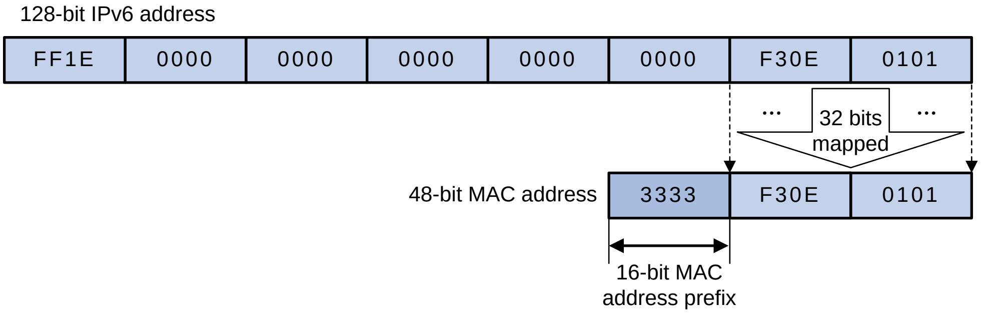


The most significant four bits of an IPv4 multicast address are fixed at 1110. In an IPv4-to-MAC address mapping, five bits of the IPv4 multicast address are lost. As a result, 32 IPv4 multicast addresses are mapped to the same IPv4 multicast MAC address. A device might receive unwanted multicast data at Layer 2 processing, which needs to be filtered by the upper layer.

IPv6 multicast MAC addresses

As defined by IANA, the most significant 16 bits of an IPv6 multicast MAC address are 0x3333. The least significant 32 bits are mapped from the least significant 32 bits of an IPv6 multicast address. Therefore, the problem of duplicate IPv6-to-MAC address mapping also arises like IPv4-to-MAC address mapping.

Figure 7 IPv6-to-MAC address mapping



Multicast protocols

Multicast protocols include the following categories:

- Layer 3 and Layer 2 multicast protocols:
  - Layer 3 multicast refers to IP multicast operating at the network layer.

    Layer 3 multicast protocols—IGMP, MLD, PIM, IPv6 PIM, MSDP, MBGP, and IPv6 MBGP.
  - Layer 2 multicast refers to IP multicast operating at the data link layer.

    Layer 2 multicast protocols—IGMP snooping, MLD snooping, PIM snooping, IPv6 PIM snooping, multicast VLAN, and IPv6 multicast VLAN.
- IPv4 and IPv6 multicast protocols:
  - **For IPv4 networks**—IGMP snooping, PIM snooping, multicast VLAN, IGMP, PIM, MSDP, and MBGP.
  - **For IPv6 networks**—MLD snooping, IPv6 PIM snooping, IPv6 multicast VLAN, MLD, IPv6 PIM, and IPv6 MBGP.

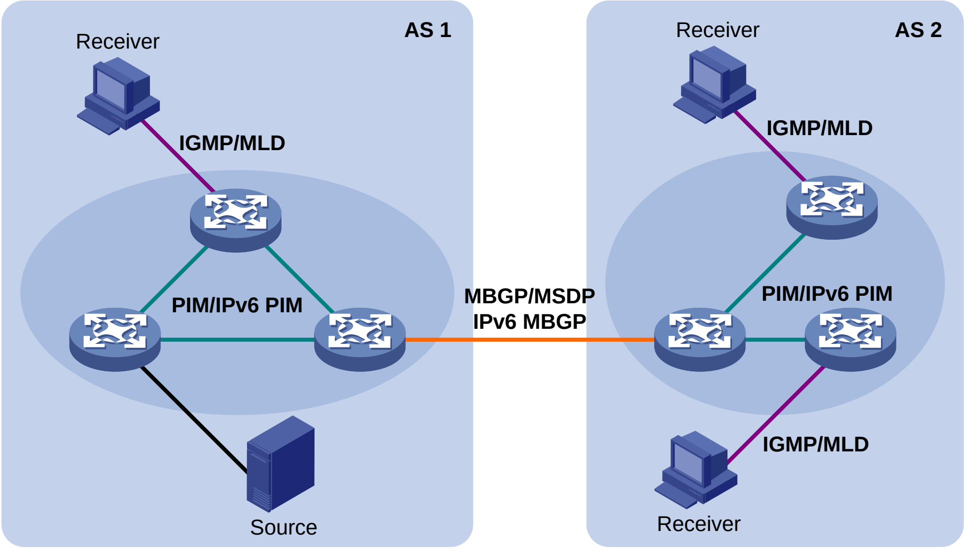
This section provides only general descriptions about applications and functions of the Layer 2 and Layer 3 multicast protocols in a network. For more information about these protocols, see the related chapters.

The device supports only the following Layer 3 multicast prototols: IGMP, MLD, and PIM.

Layer 3 multicast protocols

In Figure 8, Layer 3 multicast protocols include multicast group management protocols and multicast routing protocols.

Figure 8 Positions of Layer 3 multicast protocols



- Multicast group management protocols:

  Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) protocol are multicast group management protocols. Typically, they run between hosts and Layer 3 multicast devices that directly connect to the hosts to establish and maintain multicast group memberships.

- Multicast routing protocols:

  A multicast routing protocol runs on Layer 3 multicast devices to establish and maintain multicast routes and correctly and efficiently forward multicast packets. Multicast routes constitute loop-free data transmission paths (also known as multicast distribution trees) from a data source to multiple receivers.

  In the ASM model, multicast routes include intra-domain routes and inter-domain routes.

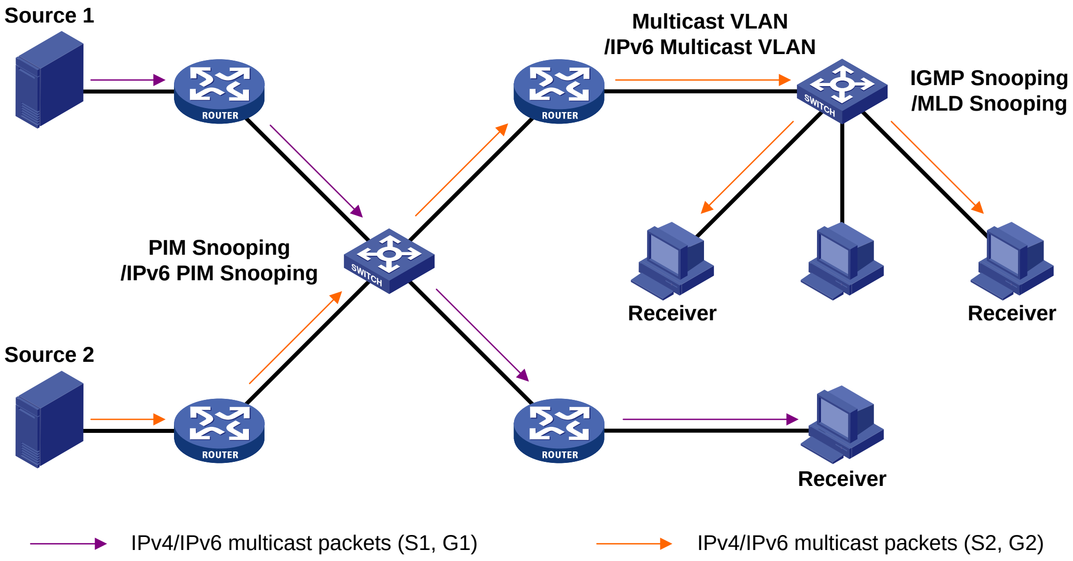
  - An intra-domain multicast routing protocol discovers multicast sources and builds multicast distribution trees within an AS to deliver multicast data to receivers. Among a variety of mature intra-domain multicast routing protocols, PIM is most widely used. Based on the forwarding mechanism, PIM has dense mode (often referred to as PIM-DM) and sparse mode (often referred to as PIM-SM).
  - An inter-domain multicast routing protocol is used for delivering multicast information between two ASs. So far, mature solutions include Multicast Source Discovery Protocol (MSDP) and MBGP. MSDP propagates multicast source information among different ASs. MBGP is an extension of the MP-BGP for exchanging multicast routing information among different ASs.

  For the SSM model, multicast routes are not divided into intra-domain routes and inter-domain routes. Because receivers know the positions of the multicast sources, channels established through PIM-SM are sufficient for the transport of multicast information.

Layer 2 multicast protocols

In Figure 9, Layer 2 multicast protocols include IGMP snooping, MLD snooping, PIM snooping, IPv6 PIM snooping, multicast VLAN, and IPv6 multicast VLAN.

Figure 9 Positions of Layer 2 multicast protocols



- IGMP snooping and MLD snooping:

  IGMP snooping and MLD snooping are multicast constraining mechanisms that run on Layer 2 devices. They manage and control multicast groups by monitoring and analyzing IGMP or MLD messages exchanged between the hosts and Layer 3 multicast devices. This effectively controls the flooding of multicast data in Layer 2 networks.

- PIM snooping and IPv6 PIM snooping:

  PIM snooping and IPv6 PIM snooping run on Layer 2 devices. They work with IGMP snooping or MLD snooping to analyze received PIM messages. Then, they add the ports that are interested in specific multicast data to a PIM snooping routing entry or IPv6 PIM snooping routing entry. In this way, multicast data can be forwarded to only the ports that are interested in the data.

- Multicast VLAN and IPv6 multicast VLAN:

  Multicast VLAN or IPv6 multicast VLAN runs on a Layer 2 device in a multicast network where multicast receivers for the same group exist in different VLANs. With these protocols, the Layer 3 multicast device sends only one copy of multicast to the multicast VLAN or IPv6 multicast VLAN on the Layer 2 device. This method avoids waste of network bandwidth and extra burden on the Layer 3 device.

Multicast packet forwarding mechanism

In a multicast model, receiver hosts of a multicast group are usually located at different areas on the network. They are identified by the same multicast group address. To deliver multicast packets to these receivers, a multicast source encapsulates the multicast data in an IP packet with the multicast group address as the destination address. Multicast routers on the forwarding paths forward multicast packets that an incoming interface receives through multiple outgoing interfaces. Compared to a unicast model, a multicast model is more complex in the following aspects:

- To ensure multicast packet transmission on the network, different routing tables are used to guide multicast forwarding. These routing tables include unicast routing tables, routing tables for multicast (for example, the MBGP routing table), and static multicast routing tables.
- To process the same multicast information from different peers received on different interfaces, the multicast device performs an RPF check on each multicast packet. The RPF check result determines whether the packet will be forwarded or discarded. The RPF check mechanism is the basis for most multicast routing protocols to implement multicast forwarding.

For more information about the RPF mechanism, see "Configuring multicast routing and forwarding" and "Configuring IPv6 multicast routing and forwarding."

IP multicast architecture

IP multicast addresses the following issues:

- Where should the multicast source transmit information to? (Multicast addressing.)
- What receivers exist on the network? (Host registration.)
- Where is the multicast source that will provide data to the receivers? (Multicast source discovery.)
- How is the information transmitted to the receivers? (Multicast routing.)

IP multicast is an end-to-end service. The multicast architecture involves the following parts:

- **Addressing mechanism**—A multicast source sends information to a group of receivers through a multicast address.
- **Host registration**—Receiver hosts can join and leave multicast groups dynamically. This mechanism is the basis for management of group memberships.
- **Multicast routing**—A multicast distribution tree (a forwarding path tree for multicast data on the network) is constructed for delivering multicast data from a multicast source to receivers.
- **Multicast applications**—A software system that supports multicast applications, such as video conferencing, must be installed on multicast sources and receiver hosts. The TCP/IP stack must support reception and transmission of multicast data.

Common notations in multicast

The following notations are commonly used in multicast transmission:

- **(*, G)**—Rendezvous point tree (RPT), or a multicast packet that any multicast source sends to multicast group G. The asterisk (*) represents any multicast source, and "G" represents a specific multicast group.
- **(S, G)**—Shortest path tree (SPT), or a multicast packet that multicast source "S" sends to multicast group "G." "S" represents a specific multicast source, and "G" represents a specific multicast group.

For more information about the concepts RPT and SPT, see "Configuring PIM."

Contents

Configuring multicast routing and forwarding

About multicast routing and forwarding

Each multicast routing protocol has its own routing table. Multicast routing information in routing entries generated by the multicast routing protocols and statically configured multicast routing entries are summarized in a set of (S, G) and (*, G) entries. All the (S, G) and (*, G) entries form a general multicast routing table. The optimal multicast routing entries in the general multicast routing table are added to the multicast forwarding table to guide multicast data forwarding.

RPF check mechanism

A multicast routing protocol uses reverse path forwarding (RPF) check to ensure the multicast data delivery along the correct path and to avoid data loops.

RPF check process

A multicast device performs the RPF check on a multicast packet as follows:

1. Chooses an optimal route back to the packet source separately from the unicast, MBGP, and static multicast routing tables.

   The term "packet source" means different things in different situations:

   - For a packet that travels along the SPT, the packet source is the multicast source.
   - For a packet that travels along the RPT, the packet source is the RP.
   - For a bootstrap message originated from the BSR, the packet source is the BSR.

   For more information about the concepts of SPT, RPT, source-side RPT, RP, and BSR, see "Configuring PIM."
2. Selects one of the three optimal routes as the RPF route as follows:
   - If the device uses the longest prefix match principle, the route with the highest subnet mask becomes the RPF route. If the routes have the same mask, the route with the highest route preference becomes the RPF route. If the routes have the same route preference, the unicast route becomes the RPF route. If equal cost routes exist, the route with the highest next hop IP address becomes the RPF route.

     For more information about the route preference, see *Layer 3—IP Routing Configuration Guide*.
   - If the device does not use the longest prefix match principle, the route with the highest route preference becomes the RPF route. If the routes have the same preference, the unicast route becomes the RPF route. If equal cost routes exist, the route with the highest next hop IP address becomes the RPF route.

   The RPF route contains the RPF interface and RPF neighbor information.

   - If the RPF route is a unicast route or MBGP route, the outgoing interface is the RPF interface and the next hop is the RPF neighbor.
   - If the RPF route is a static multicast route, the RPF interface and RPF neighbor are specified in the route.
3. Determines whether the packet arrived at the RPF interface.
   - If the packet arrived at the RPF interface, the RPF check succeeds and the packet is forwarded.

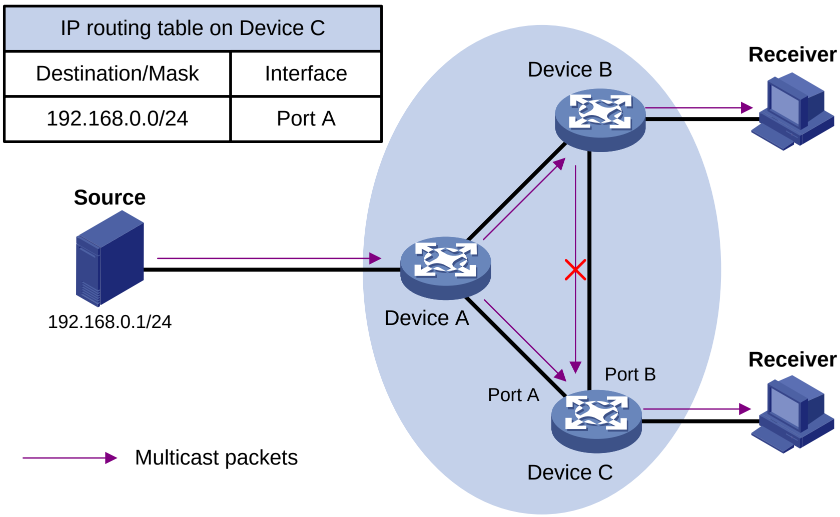
  - If the packet arrived at the non-RPF interface, the RPF check fails and the packet is discarded.

RPF check implementation in multicast

Implementing an RPF check on each received multicast packet brings a big burden to the device. The use of a multicast forwarding table is the solution to this issue. When the device creates a multicast forwarding entry for an (S, G) packet, it sets the RPF interface of the packet as the incoming interface of the (S, G) entry. After the device receives another (S, G) packet, it looks up the multicast forwarding table for a matching (S, G) entry.

- If no match is found, the device first determines the RPF route back to the packet source and the RPF interface. Then, it creates a forwarding entry with the RPF interface as the incoming interface and makes the following judgments:
  - If the receiving interface is the RPF interface, the RPF check succeeds and the device forwards the packet out of all the outgoing interfaces.
  - If the receiving interface is not the RPF interface, the RPF check fails and the device discards the packet.
- If a match is found and the matching forwarding entry contains the receiving interface, the device forwards the packet out of all the outgoing interfaces.
- If a match is found but the matching forwarding entry does not contain the receiving interface, the device determines the RPF route back to the packet source. Then, the device performs one of the following actions:
  - If the RPF interface is the incoming interface, it means that the forwarding entry is correct but the packet traveled along a wrong path. The packet fails the RPF check, and the device discards the packet.
  - If the RPF interface is not the incoming interface, it means that the forwarding entry has expired. The device replaces the incoming interface with the RPF interface and matches the receiving interface against the RPF interface. If the receiving interface is the RPF interface, the device forwards the packet out of all outgoing interfaces. Otherwise, it discards the packet.

Figure 1 RPF check process



IP routing table on Device C	
Destination/Mask	Interface
192.168.0.0/24	Port A

As shown in Figure 1, assume that unicast routes are available on the network, MBGP is not configured, and no static multicast routes have been configured on Device C. Multicast packets travel along the SPT from the multicast source to the receivers. The multicast forwarding table on Device C contains the (S, G) entry, with Port A as the incoming interface.

- If a multicast packet arrives at Device C on Port A, the receiving interface is the incoming interface of the (S, G) entry. Device C forwards the packet out of all outgoing interfaces.

- If a multicast packet arrives at Device C on Port B, the receiving interface is not the incoming interface of the (S, G) entry. Device C searches its unicast routing table and finds that the outgoing interface to the source (the RPF interface) is Port A. In this case, the (S, G) entry is correct, but the packet traveled along a wrong path. The packet fails the RPF check and Device C discards the packet.

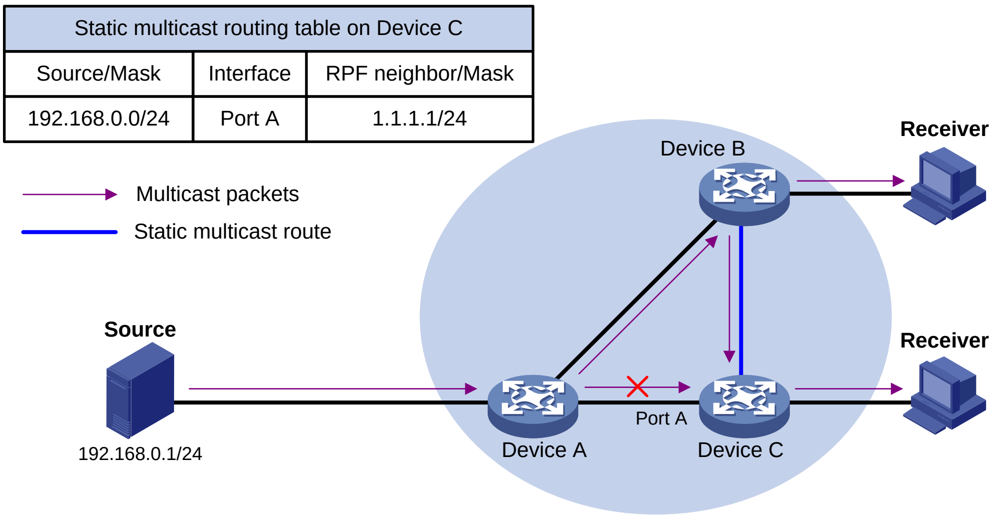
Usages of static multicast routes

A static multicast route can change an RPF route or create an RPF route.

Changing an RPF route

Typically, the topology structure of a multicast network is the same as that of a unicast network, and multicast traffic follows the same transmission path as unicast traffic does. You can configure a static multicast route for a multicast source to change the RPF route. As a result, the device creates a transmission path for multicast traffic that is different from the transmission path for unicast traffic.

Figure 2 Changing an RPF route



Static multicast routing table on Device C

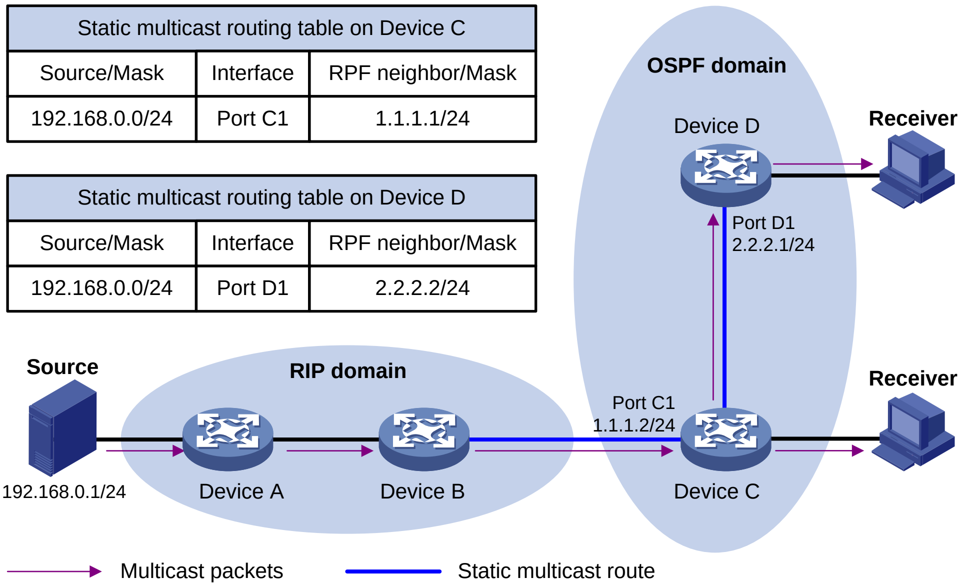
Source/Mask	Interface	RPF neighbor/Mask
192.168.0.0/24	Port A	1.1.1.1/24

As shown in Figure 2, when no static multicast route is configured, Device C's RPF neighbor on the path back to the source is Device A. The multicast data from the source travels through Device A to Device C. You can configure a static multicast route on Device C and specify Device B as Device C's RPF neighbor on the path back to the source. The multicast data from the source travels along the path: Device A to Device B and then to Device C.

Creating an RPF route

When a unicast route is blocked, multicast forwarding might be stopped due to lack of an RPF route. You can configure a static multicast route to create an RPF route. In this way, a multicast routing entry is created to guide multicast forwarding.

Figure 3 Creating an RPF route



Static multicast routing table on Device C		
Source/Mask	Interface	RPF neighbor/Mask
192.168.0.0/24	Port C1	1.1.1.1/24

Static multicast routing table on Device D		
Source/Mask	Interface	RPF neighbor/Mask
192.168.0.0/24	Port D1	2.2.2.2/24

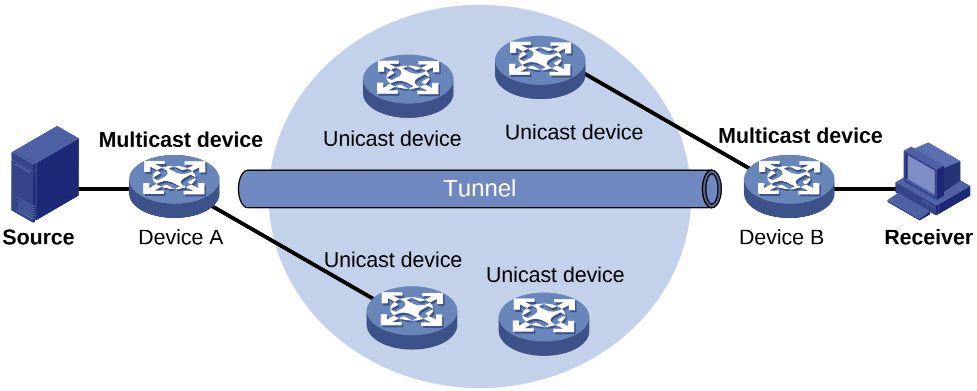
As shown in Figure 3, the RIP domain and the OSPF domain are unicast isolated from each other. For the receiver hosts in the OSPF domain to receive multicast packets from the multicast source in the RIP domain, you must configure Device C and Device D as follows:

- On Device C, configure a static multicast route for the multicast source and specify Device B as the RPF neighbor.
- On Device D, configure a static multicast route for the multicast source and specify Device C as the RPF neighbor.

Multicast forwarding across unicast subnets

Devices forward the multicast data from a multicast source hop by hop along the forwarding tree, but some devices might not support multicast protocols in a network. When the multicast data is forwarded to a device that does not support IP multicast, the forwarding path is blocked. In this case, you can enable multicast forwarding across two unicast subnets by establishing a tunnel between the devices at the edges of the two unicast subnets.

Figure 4 Multicast data transmission through a tunnel



As shown in Figure 4, a tunnel is established between Device A and Device B. Device A encapsulates the multicast data in unicast IP packets, and forwards them to Device B across the tunnel through unicast devices. Then, Device B strips off the unicast IP header and continues to forward the multicast data to the receiver.

To use this tunnel only for multicast traffic, configure the tunnel as the outgoing interface only for multicast routes.

Restrictions and guidelines: Multicast routing and forwarding configuration

The device can route and forward multicast data only through the primary IP addresses of interfaces, rather than their secondary addresses or unnumbered IP addresses. For more information about primary and secondary IP addresses, and IP unnumbered, see *Layer 3—IP Services Configuration Guide.*

Multicast routing and forwarding tasks at a glance

To configure multicast routing and forwarding, perform the following tasks:

1. Enabling IP multicast routing
2. (Optional.) Configuring static multicast routes
3. (Optional.) Specifying the longest prefix match principle
4. (Optional.) Configuring multicast load splitting
5. (Optional.) Configuring a multicast forwarding boundary
6. (Optional.) Setting the maximum number of cached unknown multicast packets
7. (Optional.) Configuring an IPv4 MVPN extranet RPF selection policy

Prerequisites for multicast routing and forwarding

Before you configure multicast routing and forwarding, configure a unicast routing protocol so that all devices in the domain can interoperate at the network layer.

Enabling IP multicast routing

About this task

Enable IP multicast routing before you configure any Layer 3 multicast functionality on the public network or VPN instance.

Procedure

1. Enter system view.

   system-view

2. Enable IP multicast routing and enter MRIB view.

   multicast routing [**vpn-instance** *vpn-instance-name*]

   By default, IP multicast routing is disabled.

Configuring static multicast routes

About this task

To configure a static multicast route for a multicast source, you can specify an RPF interface or an RPF neighbor for the multicast traffic from that source.

Restrictions and guidelines

Static multicast routes take effect only on the multicast devices on which they are configured, and will not be advertised throughout the network or redistributed to other devices.

Procedure

1. Enter system view.

   system-view

2. Configure a static multicast route.

   ip rpf-route-static [**vpn-instance** *vpn-instance-name*] *source-address* { *mask-length* | *mask* } { *rpf-nbr-address* | *interface-type interface-number* } [**preference** *preference*]

3. (Optional.) Delete all static multicast routes.

   delete ip rpf-route-static [**vpn-instance** *vpn-instance-name*]

   You can use the **undo ip rpf-route-static** command to delete a specific static multicast route or use the **delete ip rpf-route-static** command to delete all static multicast routes.

Specifying the longest prefix match principle

About this task

You can enable the device to use the longest prefix match principle for RPF route selection. For more information about RPF route selection, see "RPF check process."

Procedure

1. Enter system view.

   system-view

2. Enter MRIB view.

   multicast routing [**vpn-instance** *vpn-instance-name*]

3. Specify the longest prefix match principle.

   longest-match

   By default, the route preference principle is used.

Configuring multicast load splitting

About this task

You can enable the device to split multiple data flows on a per-source basis or on a per-source-and-group basis. This optimizes the traffic delivery.

Restrictions and guidelines

This feature does not take effect on BIDIR-PIM.

Procedure

1. Enter system view.

   system-view

2. Enter MRIB view.

   multicast routing [**vpn-instance** *vpn-instance-name*]

3. Configure multicast load splitting.

load-splitting { **source** | **source-group** }

By default, multicast load splitting is disabled.

Configuring a multicast forwarding boundary

About this task

You can configure an interface as a multicast forwarding boundary for a multicast group range. The interface cannot receive or forward multicast packets for the group range.

Restrictions and guidelines

You do not need to enable IP multicast before this configuration.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface as a multicast forwarding boundary for a multicast group range.

   multicast boundary *group-address* { *mask-length* | *mask* }

   By default, an interface is not a multicast forwarding boundary.

Setting the maximum number of cached unknown multicast packets

About this task

The device caches a multicast packet for a period of time if no matching multicast forwarding entry is found for the packet. If a multicast forwarding entry is established for the packet within the time period, the device forwards the packet. This mechanism prevents the device from mistakenly dropping multicast packets when the multicast forwarding entries for these packets are to be created.

You can set the maximum number of unknown multicast packets that can be cached for an (S, G) entry, in total, or both.

Restrictions and guidelines

As a best practice, set the value in the **multicast forwarding-table cache-unknown total** command to be far greater than the value in the **multicast forwarding-table cache-unknown per-entry** command.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of unknown multicast packets that can be cached for an (S, G) entry.

   multicast forwarding-table cache-unknown per-entry *per-entry-limit*

   By default, the device can cache only one unknown multicast packet for an (S, G) entry.

3. Set the maximum number of unknown multicast packets that can be cached in total.

   multicast forwarding-table cache-unknown total *total-limit*

   By default, the device can cache 1024 unknown multicast packets in total.

Configuring an IPv4 MVPN extranet RPF selection policy

About this task

IPv4 MVPN extranet RPF routing policies are used for IPv4 multicast transmission when multicast sources and receivers are located in different VPNs.

Restrictions and guidelines

The PIM modes in the source VPN instance and the receiver VPN instance must be the same. Only PIM-SM and PIM-SSM are supported.

Multicast packets can only be forwarded between two VPNs. The receiver VPN instance cannot also be the source VPN instance.

In PIM-SM mode, you can configure only one RPF selection policy for a multicast group in a VPN instance.

If an IPv4 MVPN extranet RPF selection policy with only the multicast group address specified is configured in the receiver VPN instance, the multicast traffic for the intra-VPN transmission will be interrupted.

To implement source-specific RPF selection in MVPN extranet, you must configure two MVPN extranet RPF routing policies as follows:

- In one policy, specify the address of the RP designated to the multicast group that requires inter-VPN multicast communication as the source address.
- In the other policy, specify the multicast source in the source VPN instance as the source address.

To implement source-and-group-specific RPF selection in MVPN extranet, you must configure two MVPN extranet RPF routing policies as follows:

- In one policy, specify the address of the RP designated to the multicast group as the source address, and specify the multicast group.
- In the other policy, specify the multicast source in the source VPN instance as the source address, and specify the multicast group.
- Make sure the multicast groups in the two policies are the same to avoid inter-VPN multicast transmission failure.

Common Layer 3 multicast supports both the source-PE-based MVPN extranet option and receiver-PE-based MVPN extranet option.

For the source-PE-based MVPN extranet option, if PIM-SM mode is used, the RP of the receiver VPN instance must be configured on the multicast source-side device.

Procedure

1. Enter system view.

   system-view

2. Enter MRIB view.

   multicast routing [**vpn-instance** *vpn-instance-name*]

3. Configure an IPv4 MVPN extranet RPF selection policy.

   multicast extranet select-rpf [**vpn-instance** *vpn-instance-name*] { **source** *source-address* { *mask* | *mask-length* } | **group** *group-address* { *mask* | *mask-length* } } *

   By default, no IPv4 MVPN extranet RPF selection policies are configured.

Display and maintenance commands for multicast routing and forwarding

CAUTION:

The **reset** commands might cause multicast data transmission failures.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about the interfaces maintained by the MRIB.	**display mrib** [**vpn-instance** *vpn-instance-name*] **interface** [*interface-type interface-number*]
Display multicast boundary information.	**display multicast** [**vpn-instance** *vpn-instance-name*] **boundary** [*group-address* [*mask-length* | *mask*]] [**interface** *interface-type interface-number*]
Display multicast fast forwarding entries.	**display multicast** [**vpn-instance** *vpn-instance-name*] **fast-forwarding cache** [*source-address* | *group-address*] * [**slot** *slot-number*]
Display DF information.	**display multicast** [**vpn-instance** *vpn-instance-name*] **forwarding df-info** [*rp-address*] [**verbose**] [**slot** *slot-number*]
Display statistics for multicast forwarding events.	**display multicast** [**vpn-instance** *vpn-instance-name*] **forwarding event** [**slot** *slot-number*]
Display multicast forwarding entries.	**display multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table** [*source-address* [**mask** { *mask-length* | *mask* }] | *group-address* [**mask** { *mask-length* | *mask* }] | **incoming-interface** *interface-type interface-number* | **outgoing-interface** { **exclude** | **include** | **match** } *interface-type interface-number* | **slot** *slot-number* | **statistics**] *
Display information about the DF list in the multicast forwarding table.	**display multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table df-list** [*group-address*] [**verbose**] [**slot** *slot-number*]
Display multicast routing entries.	**display multicast** [**vpn-instance** *vpn-instance-name*] **routing-table** [*source-address* [**mask** { *mask-length* | *mask* }] | *group-address* [**mask** { *mask-length* | *mask* }] | **incoming-interface** *interface-type interface-number* | **outgoing-interface** { **exclude** | **include** |

Task	Command
	match } *interface-type interface-number*] *
Display static multicast routing entries.	**display multicast** [**vpn-instance** *vpn-instance-name*] **routing-table static** [*source-address* { *mask-length* | *mask* }]
Display RPF information for a multicast source.	**display multicast** [**vpn-instance** *vpn-instance-name*] **rpf-info** *source-address* [*group-address*]
Clear multicast fast forwarding entries.	**reset multicast** [**vpn-instance** *vpn-instance-name*] **fast-forwarding cache** { { *source-address* | *group-address* } * | **all** } [**slot** *slot-number*]
Clear statistics for multicast forwarding events.	**reset multicast** [**vpn-instance** *vpn-instance-name*] **forwarding event**
Clear multicast forwarding entries.	**reset multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table** { { *source-address* [**mask** { *mask-length* | *mask* }] | *group-address* [**mask** { *mask-length* | *mask* }] | **incoming-interface** { *interface-type interface-number* } } * | **all** }
Clear multicast routing entries.	**reset multicast** [**vpn-instance** *vpn-instance-name*] **routing-table** { { *source-address* [**mask** { *mask-length* | *mask* }] | *group-address* [**mask** { *mask-length* | *mask* }] | **incoming-interface** *interface-type interface-number* } * | **all** }

NOTE:

- When you clear a multicast routing entry, the associated multicast forwarding entry is also cleared.
- When you clear a multicast forwarding entry, the associated multicast routing entry is also cleared.

Troubleshooting multicast routing and forwarding

Static multicast route failure

Symptom

No dynamic routing protocol is enabled on the routers, and the physical status and link layer status of interfaces are both up, but the static multicast route fails.

Solution

To resolve the problem:

1. Use the **display multicast routing-table static** command to display information about static multicast routes. Verify that the static multicast route has been correctly configured and that the route entry exists in the static multicast routing table.
2. Check the type of interface that connects the static multicast route to the RPF neighbor. If the interface is not a point-to-point interface, be sure to specify the address for the RPF neighbor.

3. If the problem persists, contact NSFOCUS Support.

Contents

Configuring IGMP

About IGMP

Internet Group Management Protocol (IGMP) establishes and maintains the multicast group memberships between a Layer 3 multicast device and the hosts on the directly connected subnet.

IGMP versions

IGMP has the following versions:

- IGMPv1 (defined by RFC 1112).
- IGMPv2 (defined by RFC 2236).
- IGMPv3 (defined by RFC 3376).

All IGMP versions support the ASM model. IGMPv3 can directly implement the SSM model. IGMPv1 and IGMPv2 must work with the IGMP SSM mapping feature to implement the SSM model. For more information about the ASM and SSM models, see "Multicast overview."

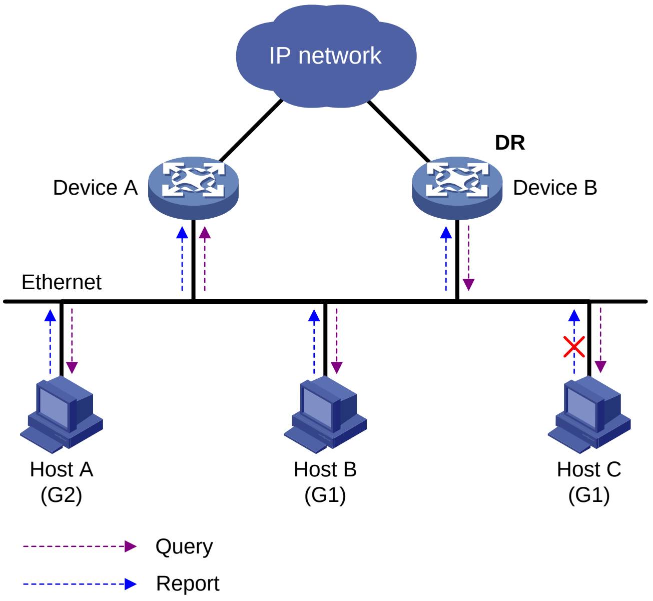
IGMPv1 overview

IGMPv1 manages multicast group memberships based on the query and response mechanism.

All devices that run IGMP on the same subnet can get IGMP membership report messages (called reports) from hosts. However, only one device can act as the IGMP querier to send IGMP query messages (called queries). The querier election mechanism determines which device acts as the IGMP querier on the subnet.

In IGMPv1, the DR elected by the multicast routing protocol (such as PIM) acts as the IGMP querier. For more information about DR, see "Configuring PIM."

Figure 1 IGMP queries and reports



As shown in Figure 1, Host B and Host C are interested in the multicast data addressed to the multicast group G1. Host A is interested in the multicast data addressed to G2. The following process describes how the hosts join the multicast groups and how the IGMP querier (Device B in Figure 1) maintains the multicast group memberships:

1. The hosts send unsolicited IGMP reports to the multicast groups they want to join without having to wait for the IGMP queries.
2. The IGMP querier periodically multicasts IGMP queries (with the destination address of 224.0.0.1) to all hosts and devices on the local subnet.
3. After receiving a query message, the host whose report delay timer expires first sends an IGMP report to multicast group G1 to announce its membership for G1. In this example, Host B sends the report message. After receiving the report from Host B, Host C suppresses its own report for G1.

   Because IGMP devices already know that G1 has a minimum of one member, other members do not need to report their memberships. This mechanism, known as the host IGMP report suppression, helps reduce traffic on the local subnet.
4. At the same time, Host A sends a report to the multicast group G2 after receiving a query.
5. Through the query and response process, the IGMP devices (Device A and Device B) determine that the local subnet has members of G1 and G2. The multicast routing protocol (PIM, for example) on the devices generates (*, G1) and (*, G2) multicast forwarding entries, where asterisk (*) represents any multicast source. These entries are the basis for subsequent multicast forwarding.
6. When the multicast data addressed to G1 or G2 reaches an IGMP device, the device looks up the multicast forwarding table. Based on the (*, G1) or (*, G2) entries, the device forwards the multicast data to the local subnet. Then, the receivers on the subnet can receive the data.

IGMPv1 does not define a leave group message (often called a leave message). When an IGMPv1 host is leaving a multicast group, it stops sending reports to that multicast group. If the subnet has no members for a multicast group, the IGMP devices will not receive any report addressed to that multicast group. In this case, the devices clear the information for that multicast group after a period of time.

IGMPv2 enhancements

Backwards-compatible with IGMPv1, IGMPv2 has introduced a querier election mechanism and a leave-group mechanism.

Querier election mechanism

In IGMPv1, the DR elected by the Layer 3 multicast routing protocol (such as PIM) acts as the querier.

IGMPv2 introduced an independent querier election mechanism. The querier election process is as follows:

1. Initially, every IGMPv2 device assumes itself to be the querier. Each device sends IGMP general query messages (called general queries) to all hosts and devices on the local subnet. The destination address is 224.0.0.1.
2. After receiving a general query, every IGMPv2 device compares the source IP address of the query with its own interface address. The device with the lowest IP address becomes the querier. All the other IGMPv2 devices become non-queriers.
3. All the non-queriers start the other querier present timer. If a device receives an IGMP query from the querier before the timer expires, it resets this timer. Otherwise, the device considers that the querier has timed out. In this case, the device initiates a new querier election process.

"Leave group" mechanism

In IGMPv1, when a host leaves a multicast group, it does not send any notification to the multicast devices. The multicast devices determine whether a group has members by using the maximum response time. This adds to the leave latency.

In IGMPv2, when a host is leaving a multicast group, the following process occurs:

1. The host sends a leave message to all devices on the local subnet. The destination address of leave messages is 224.0.0.2.
2. After receiving the leave message, the querier sends a configurable number of IGMP group-specific queries to the group that the host is leaving. Both the destination address field and the group address field of the message are the address of the multicast group that is being queried.
3. One of the remaining members (if any on the subnet) in the group should send a report within the maximum response time advertised in the group-specific queries.
4. If the querier receives a report for the group before the maximum response timer expires, it maintains the memberships for the group. Otherwise, the querier assumes that the local subnet has no member hosts for the group and stops maintaining the memberships for the group.

IGMPv3 enhancements

IGMPv3 is based on and is compatible with IGMPv1 and IGMPv2. It enhances the control capabilities of hosts and the query and report capabilities of IGMP devices.

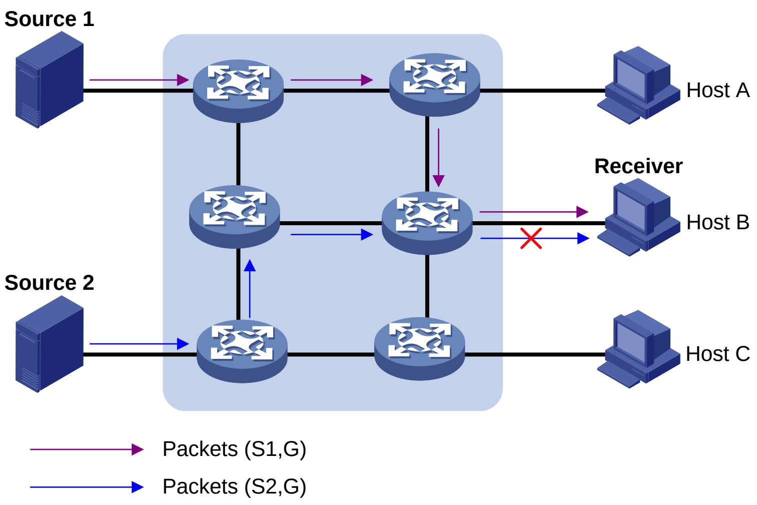
Enhancements in control capability of hosts

IGMPv3 introduced two source filtering modes (Include and Exclude). These modes allow a host to receive or reject multicast data from the specified multicast sources. When a host joins a multicast group, one of the following occurs:

- If the host expects to receive multicast data from specific sources like S1, S2, …, it sends a report with the Filter-Mode denoted as "Include Sources (S1, S2, …)."
- If the host expects to reject multicast data from specific sources like S1, S2, …, it sends a report with the Filter-Mode denoted as "Exclude Sources (S1, S2, …)."

As shown in Figure 2, the network has two multicast sources: Source 1 (S1) and Source 2 (S2). Both of these sources can send multicast data to the multicast group G. Host B wants to receive the multicast data addressed to G from Source 1 but not from Source 2.

Figure 2 Flow paths of source-and-group-specific multicast traffic



In IGMPv1 or IGMPv2, Host B cannot select multicast sources when it joins the multicast group G. The multicast streams from both Source 1 and Source 2 flow to Host B whether or not it needs them.

In IGMPv3, Host B can explicitly express that it needs to receive multicast data destined to the multicast group G from Source 1 but not from Source 2.

Enhancements in query and report capabilities

IGMPv3 introduces IGMP group-and-source queries and IGMP reports carrying group records.

- Query message carrying the source addresses

  IGMPv3 is compatible with IGMPv1 and IGMPv2 and supports IGMP general queries and IGMP group-specific queries. It also introduces IGMP group-and-source-specific queries.

  - A general query does not carry a group address or a source address.
  - A group-specific query carries a group address, but no source address.
  - A group-and-source-specific query carries a group address and one or more source addresses.

- Reports containing multiple group records

  Unlike an IGMPv1 or IGMPv2 report, an IGMPv3 report is destined to 224.0.0.22 and contains one or more group records. Each group record contains a multicast group address and a multicast source address list.

  Group records include the following categories:

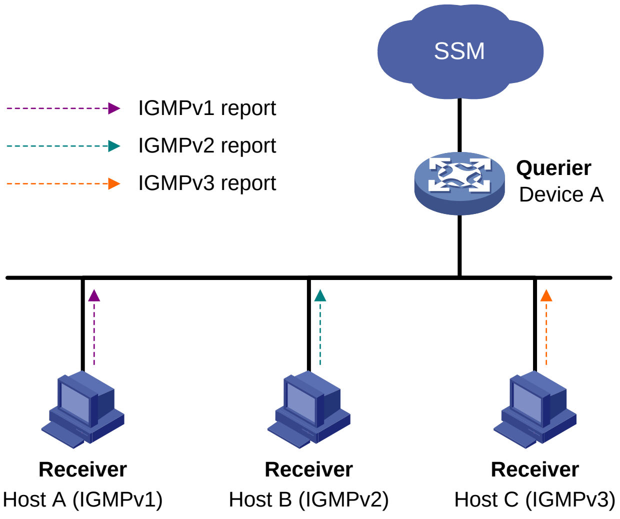
  - **IS_IN**—The current filtering mode is Include. The report sender requests the multicast data only from the sources specified in the Source Address field.
  - **IS_EX**—The current filtering mode is Exclude. The report sender requests the multicast data from any sources except those specified in the Source Address field.
  - **TO_IN**—The filtering mode has changed from Exclude to Include.
  - **TO_EX**—The filtering mode has changed from Include to Exclude.
  - **ALLOW**—The Source Address field contains a list of additional sources from which the receiver wants to obtain data. If the current filtering mode is Include, these sources are added to the multicast source list. If the current filtering mode is Exclude, these sources are deleted from the multicast source list.
  - **BLOCK**—The Source Address field contains a list of the sources from which the receiver no longer wants to obtain data. If the current filtering mode is Include, these sources are deleted from the multicast source list. If the current filtering mode is Exclude, these sources are added to the multicast source list.

IGMP SSM mapping

An IGMPv3 host can explicitly specify multicast sources in its IGMPv3 reports. From the reports, the IGMP device can obtain the multicast source addresses and directly provide the SSM service. However, an IGMPv1 or IGMPv2 host cannot specify multicast sources in its IGMPv1 or IGMPv2 reports.

The IGMP SSM mapping feature enables the IGMP device to provide SSM support for IGMPv1 or IGMPv2 hosts. The device translates (*, G) in IGMPv1 or IGMPv2 reports into (G, INCLUDE, (S1, S2...)) based on the configured IGMP SSM mappings.

Figure 3 IGMP SSM mapping



As shown in Figure 3, on an SSM network, Host A, Host B, and Host C run IGMPv1, IGMPv2, and IGMPv3, respectively. To provide the SSM service for Host A and Host B, you must configure the IGMP SSM mapping feature on Device A.

After IGMP SSM mappings are configured, Device A checks the multicast group address G in the received IGMPv1 or IGMPv2 report, and performs the following operations:

- If G is not in the SSM group range, Device A provides the ASM service.
- If G is in the SSM group range but does not match any IGMP SSM mapping, Device A drops the report.
- If G is in the SSM group range and matches IGMP SSM mappings, Device A translates (*, G) in the report into (G, INCLUDE, (S1, S2...)) to provide SSM services.

NOTE:

The IGMP SSM mapping feature does not process IGMPv3 reports.

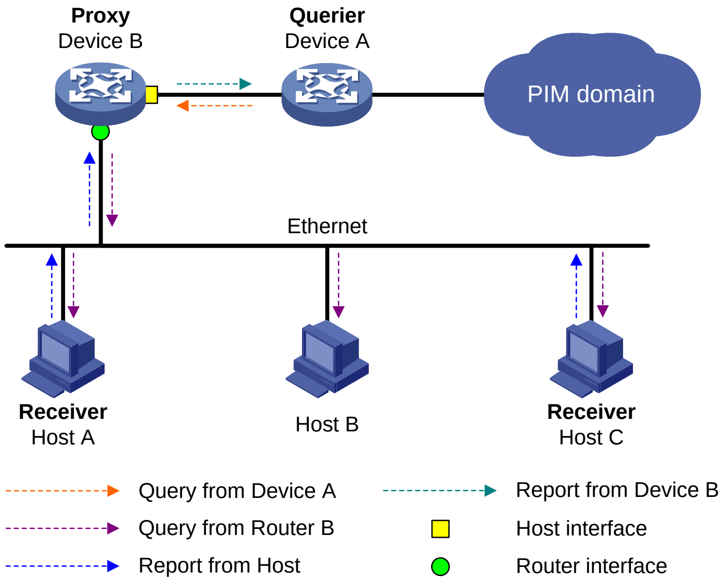
For more information about SSM group ranges, see "Configuring PIM."

IGMP proxying

As shown in Figure 4, in a simple tree-shaped topology, it is not necessary to run multicast routing protocols, such as PIM, on edge devices. Instead, you can configure IGMP proxying on these devices. With IGMP proxying configured, the edge device acts as an IGMP proxy:

- For the upstream IGMP querier, the IGMP proxy device acts as a host.
- For the downstream receiver hosts, the IGMP proxy device acts as an IGMP querier.

Figure 4 IGMP proxying



The following types of interfaces are defined in IGMP proxying:

- **Host interface**—An interface that is in the direction toward the root of the multicast forwarding tree. A host interface acts as a receiver host that is running IGMP. IGMP proxying must be enabled on this interface. This interface is also called the "proxy interface."
- **Router interface**—An interface that is in the direction toward the leaf of the multicast forwarding tree. A router interface acts as a router that is running IGMP. IGMP must be configured on this interface.

An IGMP proxy device maintains a group membership database, which stores the group memberships on all the router interfaces. The host interfaces and router interfaces perform actions based on this membership database.

- The host interfaces respond to queries according to the membership database or send join/leave messages when the database changes.
- The router interfaces participate in the querier election, send queries, and maintain memberships based on received IGMP reports.

IGMP support for VPNs

IGMP maintains group memberships on a per-interface basis. After receiving an IGMP message on an interface, IGMP processes the packet within the VPN to which the interface belongs. IGMP only communicates with other multicast protocols within the same VPN instance.

Protocols and standards

- RFC 1112, *Host Extensions for IP Multicasting*
- RFC 2236, *Internet Group Management Protocol, Version 2*
- RFC 3376, *Internet Group Management Protocol, Version 3*

IGMP tasks at a glance

To configure IGMP, perform the following tasks:

1. Enabling IGMP
2. (Optional.) Configuring basic IGMP features
   - Specifying an IGMP version
   - Configuring a static group member
   - Configuring a multicast group policy
3. (Optional.) Adjusting IGMP performance
   - Configuring IGMP query and response parameters
   - Enabling fast-leave processing
4. (Optional.) Configuring IGMP SSM mappings
5. (Optional.) Configuring IGMP proxying
   - Enabling IGMP proxying
   - Enabling multicast forwarding on a non-querier interface
   - Configuring multicast load splitting for IGMP proxy interfaces
6. (Optional.) Enabling IGMP NSR

Prerequisites for IGMP

Before you configure IGMP, you must configure any unicast routing protocol so that all devices can interoperate at the network layer.

Enabling IGMP

Restrictions and guidelines

Enable IGMP on interfaces where the multicast group memberships are established and maintained.

Procedure

1. Enter system view.
   system-view
2. Enable IP multicast routing and enter MRIB view.
   multicast routing [**vpn-instance** *vpn-instance-name*]
   By default, IP multicast routing is disabled.
   For more information about this command, see *IP Multicast Command Reference*.
3. Return to system view.
   quit
4. Enter interface view.
   interface *interface-type interface-number*
5. Enable IGMP.
   igmp enable
   By default, IGMP is disabled.

Configuring basic IGMP features

Specifying an IGMP version

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Specify an IGMP version on the interface.

```
igmp version version-number
```

By default, the IGMP version on an interface is IGMPv2.

CAUTION:

For IGMP to operate correctly, specify the same IGMP version for all devices on the same subnet.

Configuring a static group member

About this task

You can configure an interface as a static group member of a multicast group. Then, the interface can always receive multicast data addressed to the specified multicast group.

A static group member does not respond to IGMP queries. When you complete or cancel this configuration on an interface, the interface does not send an unsolicited IGMP report or leave message.

Restrictions and guidelines

The interface to be configured as a static group member has the following restrictions:

- If the interface is IGMP and PIM-SM enabled, it must be a PIM-SM DR.
- If the interface is IGMP enabled but not PIM-SM enabled, it must be an IGMP querier.

For more information about PIM-SM and DR, see "Configuring PIM."

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Configure the interface as a static group member.

```
igmp static-group group-address [ source source-address ]
```

Configuring a multicast group policy

About this task

This feature enables an interface to filter IGMP reports by using an ACL that specifies multicast groups and the optional sources. It is used to control the multicast groups that the hosts attached to an interface can join.

Restrictions and guidelines

This configuration does not take effect on static group members, because static group members do not send IGMP reports.

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.
   interface *interface-type interface-number*
3. Configure a multicast group policy.
   igmp group-policy *ipv4-acl-number* [*version-number*]

Adjusting IGMP performance

Configuring IGMP query and response parameters

About this task

The following are IGMP query and response parameters:

- **IGMP querier's robustness variable**—Number of times for retransmitting IGMP queries in case of packet loss. A higher robustness variable makes the IGMP querier more robust, but increases the timeout time for multicast groups.
- **IGMP startup query interval**—Interval at which an IGMP querier sends IGMP general queries at startup.
- **IGMP startup query count**—Number of IGMP general queries that an IGMP querier sends at startup.
- **IGMP general query interval**—Interval at which an IGMP querier sends IGMP general queries to check for multicast group members on the network.
- **IGMP last member query interval**—In IGMPv2, it sets the interval at which a querier sends group-specific queries after receiving a leave message. In IGMPv3, it sets the interval at which a querier sends group-and-source-specific queries after receiving a report that changes multicast source and group mappings.
- **IGMP last member query count**—In IGMPv2, it sets the number of group-specific queries that a querier sends after receiving a leave message. In IGMPv3, it sets the number of group-and-source-specific queries that a querier sends after receiving a report that changes multicast source and group mappings.
- **IGMP maximum response time**—Maximum time before a receiver responds with a report to an IGMP general query. This per-group timer is initialized to a random value in the range of 0 to the maximum response time specified in the IGMP query. When the timer value for a group decreases to 0, the receiver sends an IGMP report to the group.
- **IGMP other querier present timer**—Lifetime for an IGMP querier after a non-querier receives an IGMP general query. If the non-querier does not receive a new query when this timer expires, the non-querier considers that the querier has failed and starts a new querier election.

Restrictions and guidelines

To avoid frequent IGMP querier changes, set the IGMP other querier present timer greater than the IGMP general query interval. In addition, configure the same IGMP other querier present timer for all IGMP devices on the same subnet.

To avoid mistakenly deleting multicast receivers, set the IGMP general query interval greater than the maximum response time for IGMP general queries.

To speed up the response to IGMP queries and avoid simultaneous timer expirations that cause IGMP report traffic bursts, set an appropriate maximum response time.

- For IGMP general queries, the maximum response time is set by the **max-response-time** command.
- For IGMP group-specific queries and IGMP group-and-source-specific queries, the maximum response time equals the IGMP last member query interval.

The following configurations take effect only on the devices that run IGMPv2 and IGMPv3:

- Maximum response time for IGMP general queries.
- IGMP last member query interval.
- IGMP last member query count.
- IGMP other querier present interval.

You can configure the IGMP query and response parameters globally for all interfaces in IGMP view or for an interface in interface view. The interface-specific configuration takes priority over the global configuration.

Configuring the IGMP query and response parameters globally

1. Enter system view.

   system-view

2. Enter IGMP view.

   igmp [**vpn-instance** *vpn-instance-name*]

3. Set the IGMP querier's robustness variable.

   robust-count *count*

   By default, the IGMP querier's robustness variable is 2.

4. Set the IGMP startup query interval.

   startup-query-interval *interval*

   By default, the IGMP startup query interval is equal to one quarter of the IGMP general query interval.

5. Set the IGMP startup query count.

   startup-query-count *count*

   By default, the IGMP startup query count is equal to the IGMP querier's robustness variable.

6. Set the IGMP general query interval.

   query-interval *interval*

   By default, the IGMP general query interval is 125 seconds.

7. Set the IGMP last member query interval.

   last-member-query-interval *interval*

   By default, the IGMP last member query interval is 1 second.

8. Set the IGMP last member query count.

   last-member-query-count *count*

   By default, the IGMP last member query count is equal to the IGMP querier's robustness variable.

9. Set the maximum response time for IGMP general queries.

   max-response-time *time*

   By default, the maximum response time for IGMP general queries is 10 seconds.

10. Set the IGMP other querier present timer.

    other-querier-present-interval *interval*

    By default, the IGMP other querier present timer is calculated by using the following formula:
    [IGMP general query interval] × [IGMP robustness variable] + [maximum response time for IGMP general queries] / 2.

Configuring the IGMP query and response parameters on an interface

1. Enter system view.

   system-view

2. Enter interface view.

interface *interface-type interface-number*

3. Set the IGMP querier's robustness variable.

   igmp robust-count *count*

   By default, the IGMP querier's robustness variable is 2.

4. Set the IGMP startup query interval.

   igmp startup-query-interval *interval*

   By default, the IGMP startup query interval is equal to one quarter of the IGMP general query interval.

5. Set the IGMP startup query count.

   igmp startup-query-count *count*

   By default, the IGMP startup query count is equal to the IGMP querier's robustness variable.

6. Set the IGMP general query interval.

   igmp query-interval *interval*

   By default, the IGMP general query interval is 125 seconds.

7. Set the IGMP last member query interval.

   igmp last-member-query-interval *interval*

   By default, the IGMP last member query interval is 1 second.

8. Set the IGMP last member query count.

   igmp last-member-query-count *count*

   By default, the IGMP last member query count is equal to the IGMP querier's robustness variable.

9. Set the maximum response time for IGMP general queries.

   igmp max-response-time *time*

   By default, the maximum response time for IGMP general queries is 10 seconds.

10. Set the IGMP other querier present timer.

    igmp other-querier-present-interval *interval*

    By default, the IGMP other querier present timer is calculated by using the following formula:
    [IGMP general query interval] × [IGMP robustness variable] + [maximum response time for IGMP general queries] / 2.

Enabling fast-leave processing

About this task

This feature enables an IGMP querier to send leave notifications to the upstream without sending group-specific or group-and-source-specific queries after receiving a leave message. Use this feature to reduce leave latency and to preserve the network bandwidth.

Restrictions and guidelines

The fast-leave processing configuration takes effect only when the device runs IGMPv2 or IGMPv3.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable fast-leave processing.

   igmp fast-leave [**group-policy** *ipv4-acl-number*]

By default, fast-leave processing is disabled.

Configuring IGMP SSM mappings

About this task

This feature enables the device to provide SSM services for IGMPv1 or IGMPv2 hosts.

Restrictions and guidelines

This feature does not process IGMPv3 messages. Enable IGMPv3 on the receiver-side interface to ensure that IGMPv3 reports can be processed..

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter IGMP view.

   ```
   igmp [ vpn-instance vpn-instance-name ]
   ```

3. Configure an IGMP SSM mapping.

   ```
   ssm-mapping source-address ipv4-acl-number
   ```

Configuring IGMP proxying

Prerequisites for configuring IGMP proxying

Before you configure IGMP proxying, determine the router interfaces and host interfaces based on the network topology, and enable IGMP on the router interfaces.

Enabling IGMP proxying

Restrictions and guidelines

Enable IGMP proxying on the receiver-side interfaces.

On an interface enabled with IGMP proxying, only the **igmp version** command takes effect and other IGMP commands do not take effect.

If you enable both IGMP proxying and a multicast routing protocol (such as PIM) on the same device, the multicast routing protocol does not take effect.

In IGMPv1, the DR is elected by PIM and acts as the IGMP querier. Because PIM does not take effect on a proxy device, a router interface running IGMPv1 cannot be elected as the DR. To ensure that the downstream receiver hosts on the router interface can receive multicast data, you must enable multicast forwarding on the interface. For more information, see "Enabling multicast forwarding on a non-querier interface."

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable IGMP proxying.

   ```
   igmp proxying enable
   ```

By default, IGMP proxying is disabled.

Enabling multicast forwarding on a non-querier interface

About this task

Typically, only IGMP queriers can forward multicast traffic and non-queriers cannot. This prevents multicast data from being repeatedly forwarded. If a router interface on an IGMP proxy device failed in the querier election, enable multicast forwarding on the interface to forward multicast data to attached receiver hosts.

Restrictions and guidelines

A shared-media network might have multiple IGMP proxies, including one proxy acting as a querier. To avoid duplicate multicast traffic, do not enable multicast forwarding on any of the non-querier IGMP proxies for the network.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable multicast forwarding on the interface.

   `igmp proxy forwarding`

   By default, multicast forwarding is disabled on a non-querier interface.

Configuring multicast load splitting for IGMP proxy interfaces

About this task

If multiple IGMP proxy interfaces exist on the device, only the proxy interface with the highest IP address forwards multicast traffic. You can enable multicast load splitting for IGMP proxy interfaces so that all the proxy interfaces can share multicast traffic.

Procedure

1. Enter system view.

   `system-view`

2. Enter IGMP view.

   `igmp` [**`vpn-instance`** *`vpn-instance-name`*]

3. Enable multicast load splitting for IGMP proxy interfaces.

   `proxy multipath`

   By default, multicast load splitting is disabled for IGMP proxy interfaces.

Enabling IGMP NSR

About this task

This feature backs up information about IGMP interfaces and IGMP multicast groups to the standby process. The device recovers the information without cooperation of other devices when an active/standby switchover occurs. Use this feature to prevent an active/standby switchover from affecting the multicast service.

Procedure

1. Enter system view.
   system-view
2. Enable IGMP NSR.
   igmp non-stop-routing
   By default, IGMP NSR is disabled.

Display and maintenance commands for IGMP

△ **CAUTION:**

The **reset igmp group** command might cause multicast data transmission failures.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about IGMP multicast groups.	**display igmp** [**vpn-instance** *vpn-instance-name*] **group** [*group-address* | **interface** *interface-type interface-number*] [**static** | **verbose**]
Display IGMP information for interfaces.	**display igmp** [**vpn-instance** *vpn-instance-name*] **interface** [*interface-type interface-number*] [**proxy**] [**verbose**]
Display multicast group membership information maintained by the IGMP proxy.	**display igmp** [**vpn-instance** *vpn-instance-name*] **proxy group** [*group-address* | **interface** *interface-type interface-number*] [**verbose**]
Display multicast routing entries maintained by the IGMP proxy.	**display igmp** [**vpn-instance** *vpn-instance-name*] **proxy routing-table** [*source-address* [**mask** { *mask-length* | *mask* }] | *group-address* [**mask** { *mask-length* | *mask* }]] * [**verbose**]
Display IGMP SSM mappings.	**display igmp** [**vpn-instance** *vpn-instance-name*] **ssm-mapping** *group-address*
Clear dynamic IGMP multicast group entries.	**reset igmp** [**vpn-instance** *vpn-instance-name*] **group** { **all** | **interface** *interface-type interface-number* { **all** | *group-address* [**mask** { *mask* | *mask-length* }] [*source-address* [**mask** { *mask* | *mask-length* }]] } }

Troubleshooting IGMP

No membership information on the receiver-side device

Symptom

When a host sends a report for joining multicast group G, no membership information of multicast group G exists on the device closest to that host.

Solution

To resolve the problem:

1. Use the **`display igmp interface`** command to verify that the networking, interface connection, and IP address configuration are correct.
2. Use the **`display current-configuration`** command to verify that multicast routing is enabled. If it is not enabled, use the **`multicast routing`** command in system view to enable IP multicast routing. In addition, verify that IGMP is enabled on the associated interfaces.
3. Use the **`display igmp interface`** command to verify that the IGMP version on the interface is lower than that on the host.
4. Use the **`display current-configuration interface`** command to verify that no multicast group policies have been configured to filter IGMP reports for multicast group G.
5. If the problem persists, contact NSFOCUS Support.

Inconsistent membership information on the devices on the same subnet

Symptom

Different memberships are maintained on different IGMP devices on the same subnet.

Solution

To resolve the problem:

1. Use the **`display current-configuration`** command to verify the IGMP information on the interfaces. Make sure the devices on the subnet have the same IGMP settings on their interfaces.
2. Use the **`display igmp interface`** command on all devices on the same subnet to verify the IGMP-related timer settings. Make sure the settings are consistent on all the devices.
3. Use the **`display igmp interface`** command to verify that all devices on the same subnet are running the same IGMP version.
4. If the problem persists, contact NSFOCUS Support.

Contents

PIM overview

Protocol Independent Multicast (PIM) provides IP multicast forwarding by leveraging unicast static routes or unicast routing tables generated by any unicast routing protocol. PIM uses the underlying unicast routing to generate a multicast routing table without relying on any particular unicast routing protocol. PIM uses the RPF mechanism to implement multicast forwarding. For more information about RPF, see "Configuring multicast routing and forwarding."

PIM modes

Based on the implementation mechanism, PIM includes the following modes:

- Protocol Independent Multicast–Dense Mode (PIM-DM).
- Protocol Independent Multicast–Sparse Mode (PIM-SM).
- Bidirectional Protocol Independent Multicast (BIDIR-PIM).
- Protocol Independent Multicast Source-Specific Multicast (PIM-SSM).

In this document, a PIM domain refers to a network that contains PIM devices.

PIM-DM

PIM-DM uses the push mode for multicast forwarding, and is suitable for small-sized networks with densely distributed multicast members.

PIM-DM assumes that all downstream nodes want to receive multicast data from a source, so multicast data is flooded to all downstream nodes on the network. Branches without downstream receivers are pruned from the forwarding trees. When a pruned branch has new receivers, the graft mechanism turns the pruned branch into a forwarding branch.

In PIM-DM, the multicast forwarding paths for a multicast group constitutes a forwarding tree. The forwarding tree is rooted at the multicast source and has multicast group members as its "leaves." Because the forwarding tree consists of the shortest paths from the multicast source to the receivers, it is also called a "shortest path tree (SPT)."

PIM-DM mechanisms include neighbor discovery, SPT building, graft, and assert.

Neighbor discovery

In a PIM domain, each PIM interface on a device periodically multicasts PIM hello messages to all other PIM devices (identified by the address 224.0.0.13) on the local subnet. Through the exchanging of hello messages, all PIM devices on the subnet determine their PIM neighbors, maintain PIM neighboring relationship with other devices, and build and maintain SPTs.

SPT building

The process of building an SPT is the flood-and-prune process:

1. In a PIM-DM domain, the multicast data from the multicast source S to the multicast group G is flooded throughout the domain. A device performs an RPF check on the multicast data. If the RPF check succeeds, the device creates an (S, G) entry and forwards the data to all downstream nodes on the network. In the flooding process, all the devices in the PIM-DM domain create the (S, G) entry.
2. The nodes without downstream receivers are pruned. A device that has no downstream receivers multicasts a prune message to all PIM devices on the subnet. When an upstream

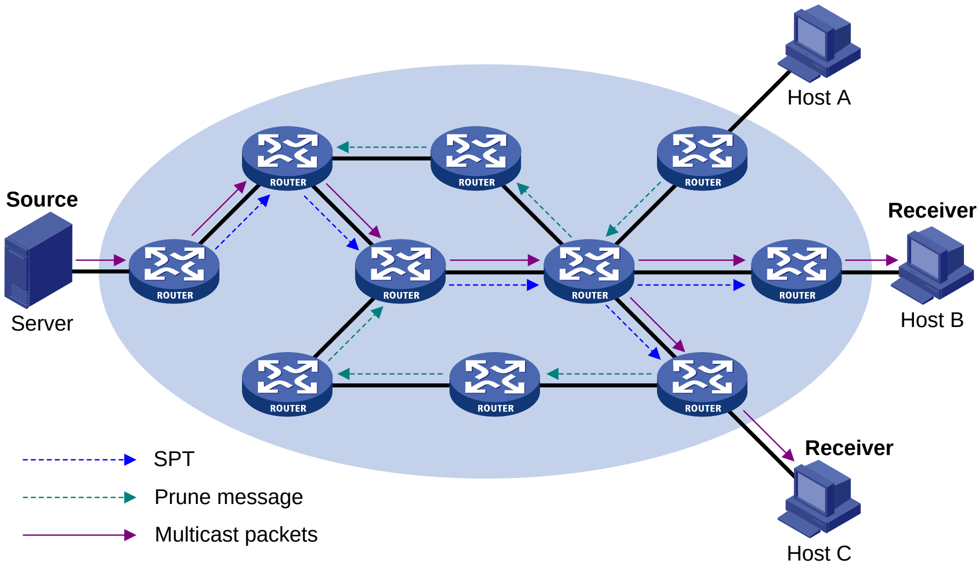
node receives the prune message, it removes the receiving interface from the (S, G) entry. In this way, the upstream stream node stops forwarding subsequent packets addressed to that multicast group down to this node.

NOTE:

An (S, G) entry contains a multicast source address S, a multicast group address G, an outgoing interface list, and an incoming interface.

A prune process is initiated by a leaf device. As shown in Figure 1, the device interface that does not have any downstream receivers initiates a prune process by sending a prune message toward the multicast source. This prune process goes on until only necessary branches are left in the PIM-DM domain, and these necessary branches constitute an SPT.

Figure 1 SPT building



The pruned state of a branch has a finite holdtime timer. When the timer expires, multicast data is again forwarded to the pruned branch. The flood-and-prune cycle takes place periodically to maintain the forwarding branches.

Graft

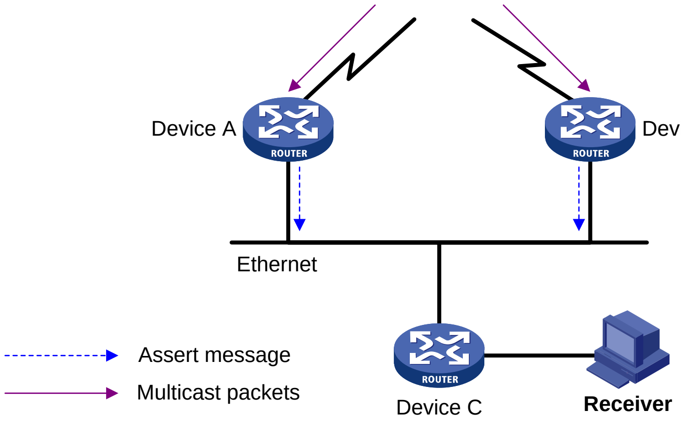
A previously pruned branch might have new downstream receivers. To reduce the latency for resuming the forwarding capability of this branch, a graft mechanism is used as follows:

1. The node that needs to receive the multicast data sends a graft message to its upstream node, telling it to rejoin the SPT.
2. After receiving this graft message on an interface, the upstream node adds the receiving interface to the outgoing interface list of the (S, G) entry. It also sends a graft-ack message to the graft sender.
3. If the graft sender receives a graft-ack message, the graft process finishes. Otherwise, the graft sender continues to send graft messages at a graft retry interval until it receives an acknowledgment from its upstream node.

Assert

On a subnet with more than one multicast device, the assert mechanism shuts off duplicate multicast flows to the network. It does this by electing a unique multicast forwarder for the subnet.

Figure 2 Assert mechanism



As shown in Figure 2, after Device A and Device B receive an (S, G) packet from the upstream node, they both forward the packet to the local subnet. As a result, the downstream node Device C receives two identical multicast packets. In addition, both Device A and Device B, on their downstream interfaces, receive a duplicate packet forwarded by the other. After detecting this condition, both devices send an assert message to all PIM devices (224.0.0.13) on the local subnet through the interface that received the packet. The assert message contains the multicast source address (S), the multicast group address (G), and the metric preference and metric of the unicast route/MBGP route/static multicast route to the multicast source. By comparing these parameters, either Device A or Device B becomes the unique forwarder of the subsequent (S, G) packets on the shared-media LAN. The comparison process is as follows:

1. The device with a higher metric preference to the multicast source wins.
2. If both devices have the same metric preference, the device with a smaller metric wins.
3. If both devices have the same metric, the device with a higher IP address on the downstream interface wins.

PIM-SM

PIM-DM uses the flood-and-prune cycles to build SPTs for multicast data forwarding. Although an SPT has the shortest paths from the multicast source to the receivers, it is built with a low efficiency. Therefore, PIM-DM is not suitable for large and medium-sized networks. PIM-SM uses the pull mode for multicast forwarding, and it is suitable for large- and medium-sized networks with sparsely and widely distributed multicast group members.

PIM-SM assumes that no hosts need multicast data. A multicast receiver must express its interest in the multicast data for a multicast group before the data is forwarded to it. A rendezvous point (RP) is the core of a PIM-SM domain. Relying on the RP, SPTs and rendezvous point trees (RPTs) are established and maintained to implement multicast data forwarding. An SPT is rooted at the multicast source and has the RPs as its leaves. An RPT is rooted at the RP and has the receiver hosts as its leaves.

PIM-SM mechanisms include neighbor discovery, DR election, RP discovery, Anycast RP, RPT building, multicast source registration, switchover to SPT, and assert.

Neighbor discovery

PIM-SM uses the same neighbor discovery mechanism as PIM-DM does. For more information, see "Neighbor discovery."

DR election

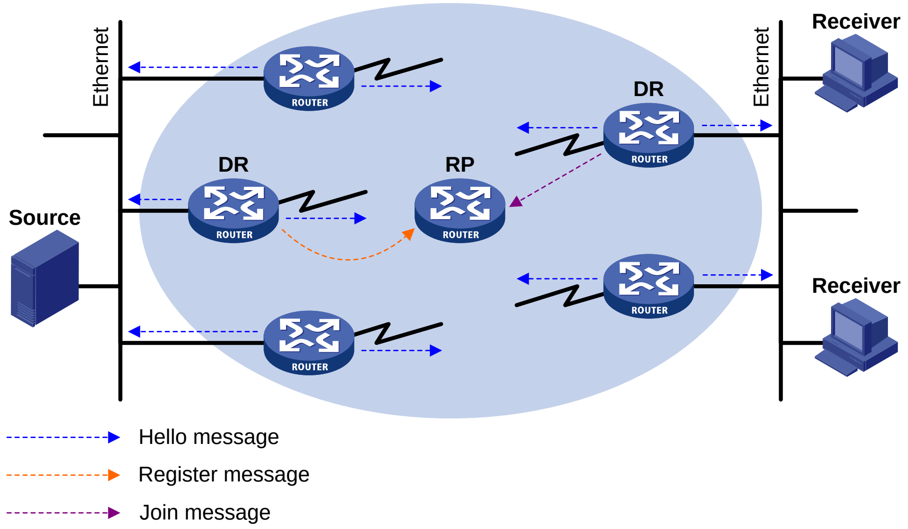
A designated router (DR) is required on both the source-side network and receiver-side network. A source-side DR acts on behalf of the multicast source to send register messages to the RP. The receiver-side DR acts on behalf of the multicast receivers to send join messages to the RP.

PIM-DM does not require a DR. However, if IGMPv1 runs on any shared-media LAN in a PIM-DM domain, a DR must be elected to act as the IGMPv1 querier for the LAN. For more information about IGMP, see "Configuring IGMP."

IMPORTANT:

IGMP must be enabled on the device that acts as the receiver-side DR. Otherwise, the receiver hosts attached to the DR cannot join any multicast groups.

Figure 3 DR election



As shown in Figure 3, the DR election process is as follows:

1. The devices on the shared-media LAN send hello messages to one another. The hello messages contain the DR priority for DR election. The device with the highest DR priority is elected as the DR.
2. The device with the highest IP address wins the DR election under one of following conditions:
   - All the devices have the same DR election priority.
   - A device does not support carrying the DR priority in hello messages.

If the DR fails, its PIM neighbor lifetime expires and the other devices will initiate to elect a new DR.

RP discovery

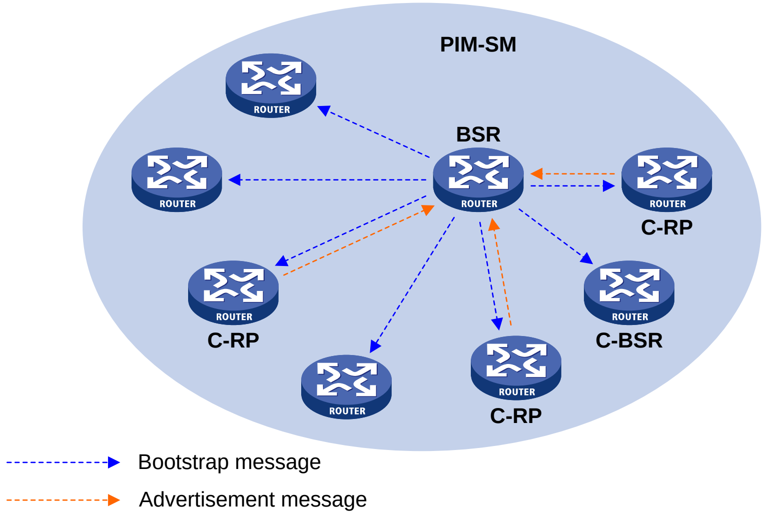
An RP is the core of a PIM-SM domain. A multicast group can have only one RP for multicast forwarding, and an RP can be designated to multiple multicast groups. RPs can be statically configured or dynamically elected through bootstrap router (BSR) mechanism. The BSM mechanism lessens the RP burden and optimizes the topological structure of the RPT.

BSR mechanism includes the following roles:

- **Candidate-RPs (C-RPs)**—An RP is dynamically elected from C-RPs to provide services to a multicast group.

- **BSR**—A BSR is the core of the administrative core of the PIM-SM domain. It is responsible for collecting and advertising RP information in the whole domain. A PIM-SM domain has only one BSR, and the BSR is elected from C-BSRs.
- **Candidate-BSRs (C-BSRs)**—Any devices in the PIM-SM domain can act as C-BSRs and the BSR is elected from the C-BSRs. Once the BSR fails, a new BSR is elected from the C-BSRs to avoid multicast traffic interruption.

Figure 4 Information exchange between C-RPs and BSR



As shown in Figure 4, an RP is elected as follows:

1. Each C-BSR sends a bootstrap message (BSM) to other devices in the PIM-SM domain.
2. When a C-BSR receives a BSM from another C-BSR, it compares its own priority with the priority carried in the message. The C-BSR with a higher priority wins the BSR election. If a tie exists in the priority, the C-BSR with a higher IP address wins. The loser uses the winner's BSR address to replace its own BSR address and no longer regards itself as the BSR. The winner retains its own BSR address and continues to regard itself as the BSR.
3. Each C-RP periodically unicasts an advertisement message to the BSR. An advertisement message contains the address of the advertising C-RP and the multicast group range to which it is designated.
4. The BSR collects these advertisement messages and organizes the C-RP information into an RP-set, which is a database of mappings between multicast groups and RPs. The BSR encapsulates the RP-set information in the bootstrap messages (BSMs) and floods the BSMs to the entire PIM-SM domain.
5. All devices in the PIM-SM domain select an RP for a multicast group based on the following rules:
    a. The C-RP that is designated to the smallest multicast group range wins.
    b. If the C-RPs are designated to the same group ranges, the C-RP with the highest priority wins.
    c. If the C-RPs have the same priority, the C-RP with the largest hash value wins. The hash value is calculated through the hash algorithm.
    d. If the C-RPs have the same hash value, the C-RP with the highest IP address wins.

Anycast RP

PIM-SM requires only one active RP to serve each multicast group. If the active RP fails, the multicast traffic might be interrupted. The Anycast RP mechanism enables redundancy backup among RPs by configuring multiple RPs with the same IP address. A multicast source registers with the closest RP or a receiver joins the closest RP to implement source information synchronization.

Anycast RP has the following benefits:

- **Optimal RP path**—A multicast source registers with the closest RP to build an optimal SPT. A receiver joins the closest RP to build an optimal RPT.
- **Redundancy backup among RPs**—When an RP fails, the RP-related sources will register with the closest available RPs and the receiver-side DRs will join the closest available RPs. This provides redundancy backup among RPs.

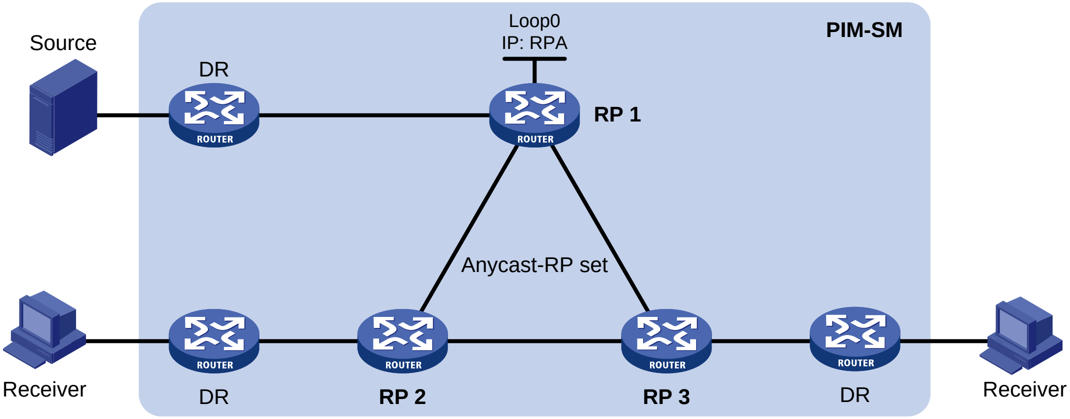
Anycast RP is implemented based on PIM-SM. You can configure multiple RPs for one multicast group and add them to an Anycast RP set.

PIM-SM-based Anycast RP introduces the following concepts:

- **Anycast RP set**—A set of RPs that are designated to the same multicast group.
- **Anycast RP member**—Each RP in the Anycast RP set.
- **Anycast RP member address**—IP address of each Anycast RP member for communication among the RP members.
- **Anycast RP address**—IP address of the Anycast RP set for communication within the PIM-SM domain. It is also known as RPA.

As shown in Figure 5, RP 1, RP 2, and RP 3 are members of an Anycast RP set.

Figure 5 Anycast RP through PIM-SM



The following describes how Anycast RP through PIM-SM is implemented:

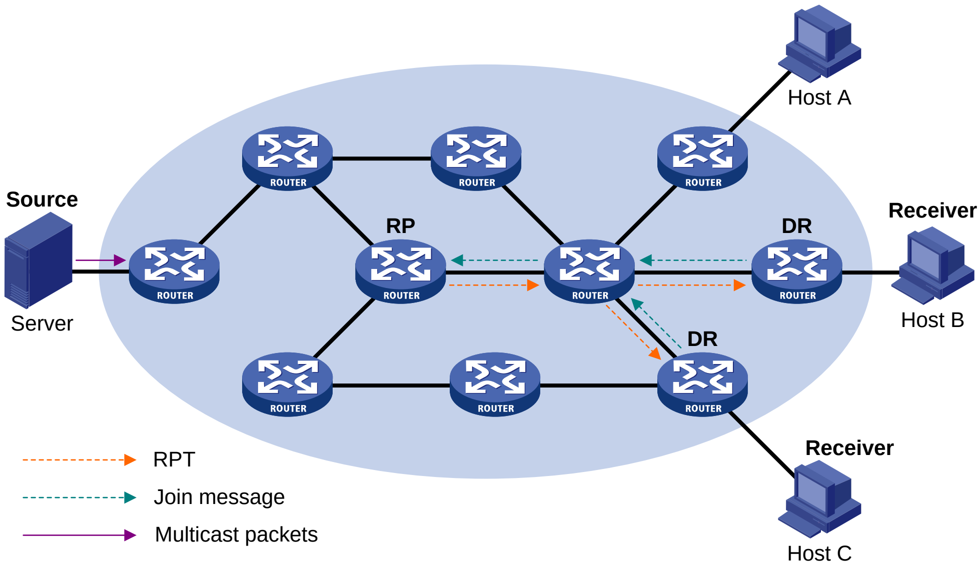
1. RP 1 receives a register message destined to RPA. Because the message is not from other Anycast RP members (RP 2 or RP 3), RP 1 considers that the register message is from the DR. RP 1 changes the source IP address of the register message to its own address and sends the message to the other members (RP 2 and RP 3).

   If a device acts as both a DR and an RP, it creates a register message, and then forwards the message to the other RP members.
2. After receiving the register message, RP 2 and RP 3 find out that the source address of the register message is an Anycast RP member address. They stop forwarding the message to other devices.

In Anycast RP implementation, an RP must forward the register message from the DR to other Anycast RP members to synchronize multicast source information.

RPT building

Figure 6 RPT building in a PIM-SM domain



As shown in Figure 6, the process of building an RPT is as follows:

1. When a receiver wants to join the multicast group G, it uses an IGMP message to inform the receiver-side DR.
2. After getting the receiver information, the DR sends a join message, which travels hop by hop to the RP for the multicast group.
3. The devices along the path from the DR to the RP form an RPT branch. Each device on this branch adds to its forwarding table a (*, G) entry, where the asterisk (*) represents any multicast source. The RP is the root of the RPT, and the DR is a leaf of the RPT.

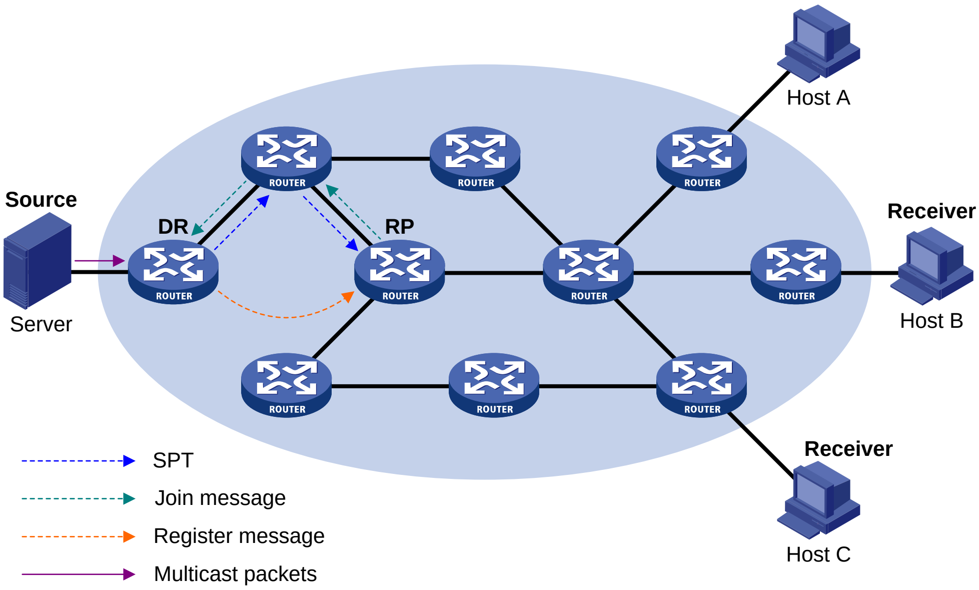
When the multicast data addressed to the multicast group G reaches the RP, the RP forwards the data to the DR along the established RPT, and finally to the receiver.

When a receiver is no longer interested in the multicast data addressed to the multicast group G, the receiver-side DR sends a prune message. The prune message goes hop by hop along the RPT to the RP. After receiving the prune message, the upstream node deletes the interface that connects to this downstream node from the outgoing interface list. At the same time, the upstream device checks for the existence of receivers for that multicast group. If no receivers for the multicast group exist, the device continues to forward the prune message to its upstream device.

Multicast source registration

The multicast source uses the registration process to inform an RP of its presence.

Figure 7 Multicast source registration



As shown in Figure 7, the multicast source registers with the RP as follows:

1. The multicast source S sends the first multicast packet to the multicast group G. When receiving the multicast packet, the source-side DR encapsulates the packet into a PIM register message and unicasts the message to the RP.
2. After the RP receives the register message, it decapsulates the register message and forwards the register message down to the RPT. Meanwhile, it sends an (S, G) source-specific join message toward the multicast source. The devices along the path from the RP to the multicast source constitute an SPT branch. Each device on this branch creates an (S, G) entry in its forwarding table.
3. The subsequent multicast data from the multicast source are forwarded to the RP along the established SPT. When the multicast data reaches the RP along the SPT, the RP forwards the data to the receivers along the RPT. Meanwhile, it unicasts a register-stop message to the source-side DR to prevent the DR from unnecessarily encapsulating the data.

Switchover to SPT

This section takes a router as an example. As compared with the switchover to SPT on a router, switchover to SPT is implemented in a simpler way on a switch.

In a PIM-SM domain, only one RP and one RPT provide services for a specific multicast group. Before the switchover to SPT occurs, the source-side DR encapsulates all multicast data in register messages and sends them to the RP. After receiving these register messages, the RP decapsulates them and forwards them to the receiver-side DR along the RPT.

Multicast forwarding along the RPT has the following weaknesses:

- Encapsulation and decapsulation are complex on the source-side DR and the RP.
- The path for a multicast packet might not be the shortest one.
- The RP might be overloaded by multicast traffic bursts.

To eliminate these weaknesses, PIM-SM allows an RP or the receiver-side DR to initiate the switchover to SPT when the traffic rate exceeds a specific threshold.

- The RP initiates the switchover to SPT:

  The RP periodically checks the multicast packet forwarding rate. If the RP finds that the traffic rate exceeds the specified threshold, it sends an (S, G) source-specific join message toward the multicast source. The routers along the path from the RP to the multicast source constitute an SPT. The subsequent multicast data is forwarded to the RP along the SPT without being encapsulated into register messages.

For more information about the switchover to SPT initiated by the RP, see "Multicast source registration."

- The receiver-side DR initiates the switchover to SPT:

  The receiver-side DR periodically checks the forwarding rate of the multicast packets that the multicast source S sends to the multicast group G. If the forwarding rate exceeds the specified threshold, the DR initiates the switchover to SPT as follows:

  a. The receiver-side DR sends an (S, G) source-specific join message toward the multicast source. The routers along the path create an (S, G) entry in their forwarding table to constitute an SPT branch.

  b. When the multicast packets reach the router where the RPT and the SPT branches, the router drops the multicast packets that travel along the RPT. It then sends a prune message with the RP bit toward the RP.

  c. After receiving the prune message, the RP forwards it toward the multicast source (supposed only one receiver exists). Thus, the switchover to SPT is completed. The subsequent multicast packets travel along the SPT from the multicast source to the receiver hosts.

With the switchover to SPT, PIM-SM builds SPTs more economically than PIM-DM does.

Assert

PIM-SM uses a similar assert mechanism as PIM-DM does. For more information, see "Assert."

BIDIR-PIM

In some many-to-many applications, such as a multi-side video conference, multiple receivers of a multicast group might be interested in the multicast data from multiple multicast sources. With PIM-DM or PIM-SM, each device along the SPT must create an (S, G) entry for each multicast source, consuming a lot of system resources.

BIDIR-PIM addresses the problem. Derived from PIM-SM, BIDIR-PIM builds and maintains a bidirectional RPT, which is rooted at the RP and connects the multicast sources and the receivers. Along the bidirectional RPT, the multicast sources send multicast data to the RP, and the RP forwards the data to the receivers. Each device along the bidirectional RPT needs to maintain only one (*, G) entry, saving system resources.

BIDIR-PIM is suitable for a network with dense multicast sources and receivers.

BIDIR-PIM mechanisms include neighbor discovery, RP discovery, DF election, and bidirectional RPT building.

Neighbor discovery

BIDIR-PIM uses the same neighbor discovery mechanism as PIM-SM does. For more information, see "Neighbor discovery."

RP discovery

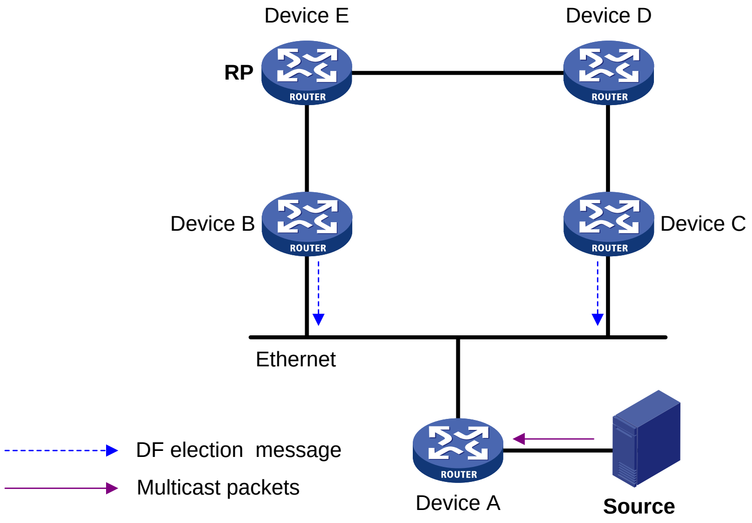
BIDIR-PIM uses the same RP discovery mechanism as PIM-SM does. For more information, see "RP discovery." In BIDIR-PIM, an RPF interface is the interface toward an RP, and an RPF neighbor is the address of the next hop to the RP.

DF election

On a subnet with multiple multicast devices, duplicate multicast packets might be forwarded to the RP. To address this issue, BIDIR-PIM uses a designated forwarder (DF) election mechanism to elect a unique DF for each RP on a subnet. Only the DFs can forward multicast data to the RP.

DF election is not necessary for an RPL.

Figure 8 DF election



As shown in Figure 8, without the DF election mechanism, both Device B and Device C can receive multicast packets from Route A. They also can forward the packets to downstream devices on the local subnet. As a result, the RP (Device E) receives duplicate multicast packets.

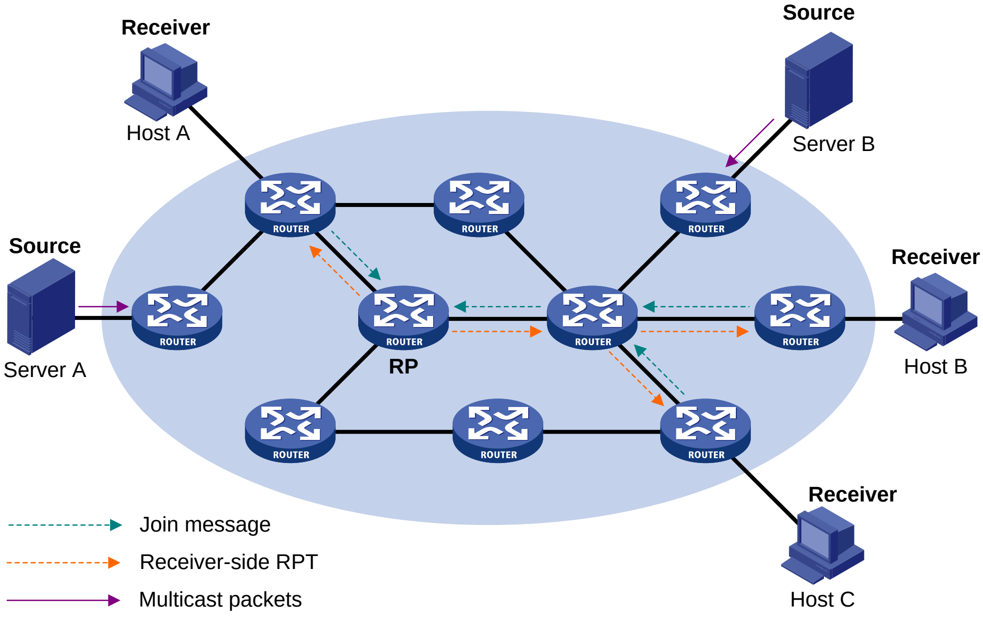
With the DF election mechanism, once receiving the RP information, Device B and Device C multicast a DF election message to all PIM devices (224.0.0.13) to initiate a DF election process. The election message carries the RP's address, and the route preference and the metric of the unicast route or static multicast route to the RP. A DF is elected as follows:

1. The device with a higher route preference becomes the DF.
2. If the devices have the same route preference, the device with a lower metric becomes the DF.
3. If the devices have the same metric, the device with a higher IP address becomes the DF.

Bidirectional RPT building

A bidirectional RPT comprises a receiver-side RPT and a source-side RPT. The receiver-side RPT is rooted at the RP and takes the devices that directly connect to the receivers as leaves. The source-side RPT is also rooted at the RP but takes the devices that directly connect to the sources as leaves. The processes for building these two RPTs are different.

Figure 9 RPT building at the receiver side

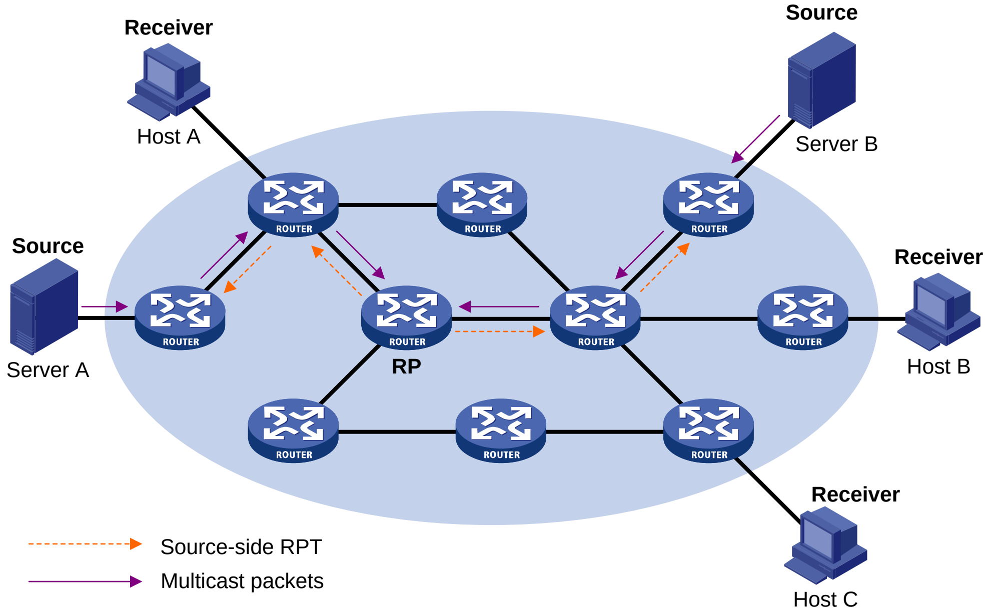


As shown in Figure 9, the process for building a receiver-side RPT is the same as the process for building an RPT in PIM-SM:

1. When a receiver wants to join the multicast group G, it uses an IGMP message to inform the directly connected device.
2. After receiving the message, the device sends a join message, which is forwarded hop by hop to the RP for the multicast group.
3. The devices along the path from the receiver's directly connected device to the RP form an RPT branch. Each device on this branch adds a (*, G) entry to its forwarding table.

After a receiver host leaves the multicast group G, the directly connected device multicasts a prune message to all PIM devices on the subnet. The prune message goes hop by hop along the reverse direction of the RPT to the RP. After receiving the prune message, an upstream node removes the interface that connects to the downstream node from the outgoing interface list. At the same time, the upstream device checks the existence of receivers for that multicast group. If no receivers for the multicast group exist, the device continues to forward the prune message to its upstream device.

Figure 10 RPT building at the multicast source side



As shown in Figure 10, the process for building a source-side RPT is relatively simple:

1. When a multicast source sends multicast packets to the multicast group G, the DF in each subnet unconditionally forwards the packets to the RP.
2. The devices along the path from the source's directly connected device to the RP constitute an RPT branch. Each device on this branch adds to its forwarding table a (*, G) entry.

After a bidirectional RPT is built, the multicast sources send multicast traffic to the RP along the source-side RPT. Then, the RP forwards the traffic to the receivers along the receiver-side RPT.

IMPORTANT:

If a receiver and a multicast source are at the same side of the RP, the source-side RPT and the receiver-side RPT might meet at a node before reaching the RP. In this case, the multicast packets from the multicast source to the receiver are directly forwarded by the node, instead of by the RP.

Administrative scoping

Typically, a PIM-SM domain or a BIDIR-PIM domain contains only one BSR, which is responsible for advertising RP-set information within the entire domain. The information about all multicast groups is forwarded within the network that the BSR administers. This is called the "non-scoped BSR mechanism."

Administrative scoping mechanism

To implement refined management, you can divide a PIM-SM domain or BIDIR-PIM domain into a global-scoped zone and multiple administratively-scoped zones (admin-scoped zones). This is called the "administrative scoping mechanism."

The administrative scoping mechanism effectively releases stress on the management in a single-BSR domain and enables provision of zone-specific services through private group addresses.

Admin-scoped zones are divided for multicast groups. Zone border devices (ZBRs) form the boundary of an admin-scoped zone. Each admin-scoped zone maintains one BSR for multicast groups within a specific range. Multicast protocol packets, such as assert messages and BSMs, for a specific group range cannot cross the boundary of the admin-scoped zone for the group range. Multicast group ranges that are associated with different admin-scoped zones can have intersections. However, the multicast groups in an admin-scoped zone are valid only within the local zone, and theses multicast groups are regarded as private group addresses.

The global-scoped zone maintains a BSR for the multicast groups that do not belong to any admin-scoped zones.

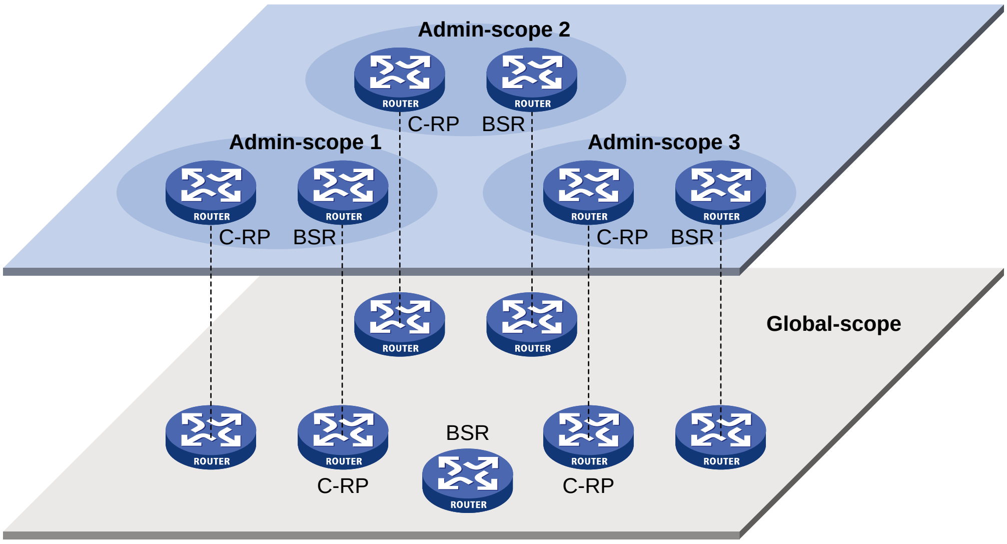
Relationship between admin-scoped zones and the global-scoped zone

The global-scoped zone and each admin-scoped zone have their own C-RPs and BSRs. These devices are effective only on their respective zones, and the BSR election and the RP election are implemented independently. Each admin-scoped zone has its own boundary. The multicast information within a zone cannot cross this boundary in either direction. You can have a better understanding of the global-scoped zone and admin-scoped zones based on geographical locations and multicast group address ranges.

- In view of geographical locations:

  An admin-scoped zone is a logical zone for particular multicast groups. The multicast packets for such multicast groups are confined within the local admin-scoped zone and cannot cross the boundary of the zone.

  Figure 11 Relationship in view of geographical locations

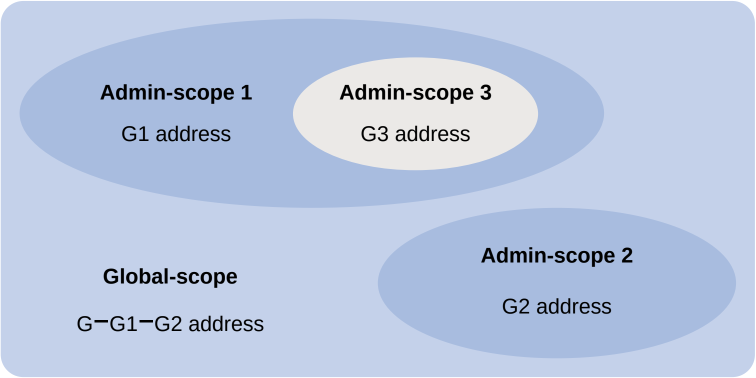


  As shown in Figure 11, for the multicast groups in a specific group address range, the admin-scoped zones must be geographically separated and isolated. A device cannot belong to multiple admin-scoped zones. An admin-scoped zone cannot contain a device that belongs to any other admin-scoped zone. However, the global-scoped zone includes all devices in the PIM-SM domain or BIDIR-PIM domain. Multicast packets that do not belong to any admin-scoped zones are forwarded in the entire PIM-SM domain or BIDIR-PIM domain.

- In view of multicast group address ranges:

  Each admin-scoped zone is designated to specific multicast groups, of which the multicast group addresses are valid only within the local zone. The multicast groups of different admin-scoped zones might have intersections. All the multicast groups other than those of the admin-scoped zones use the global-scoped zone.

Figure 12 Relationship in view of multicast group address ranges



As shown in Figure 12, the admin-scoped zones 1 and 2 have no intersection, but the admin-scoped zone 3 is a subset of the admin-scoped zone 1. The global-scoped zone provides services for all the multicast groups that are not covered by the admin-scoped zones 1 and 2, G−G1−G2 in this case.

PIM-SSM

The ASM model includes PIM-DM and PIM-SM. The SSM model can be implemented by leveraging part of the PIM-SM technique. It is also called "PIM-SSM."

The SSM model provides a solution for source-specific multicast. It maintains the relationship between hosts and devices through IGMPv3.

In actual applications, part of IGMPv3 and PIM-SM techniques are adopted to implement the SSM model. In the SSM model, because receivers have located a multicast source, no RP or RPT is required. Multicast sources do not register with the RP.

PIM-SSM mechanisms include neighbor discovery, DR election, and SPT building.

Neighbor discovery

PIM-SSM uses the same neighbor discovery mechanism as PIM-SM. For more information, see "Neighbor discovery."

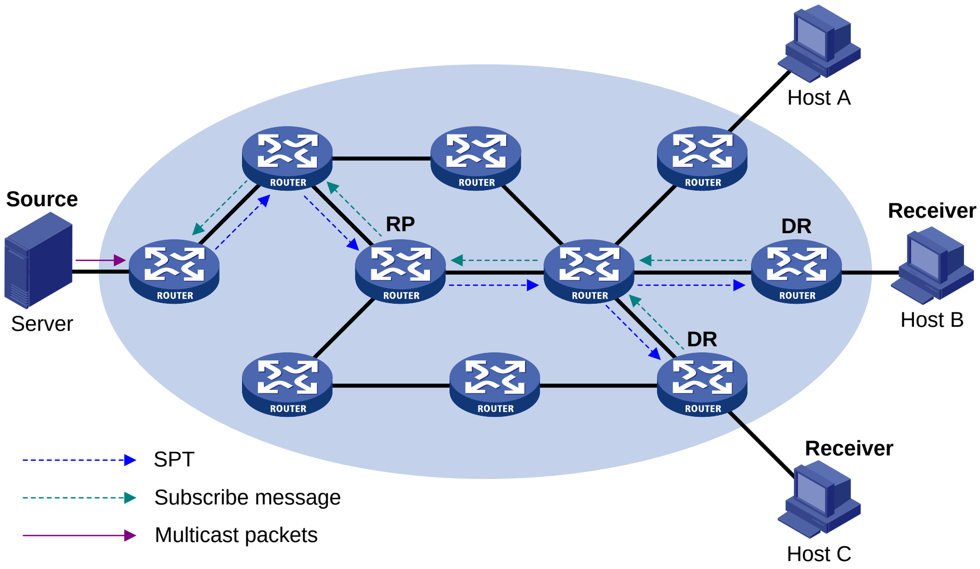
DR election

PIM-SSM uses the same DR election mechanism as PIM-SM. For more information, see "DR election."

SPT building

The decision to build an RPT for PIM-SM or an SPT for PIM-SSM depends on whether the multicast group that the receiver joins is in the SSM group range. The SSM group range reserved by IANA is 232.0.0.0/8.

Figure 13 SPT building in PIM-SSM



As shown in Figure 13, Host B and Host C are receivers. They send IGMPv3 report messages to their DRs to express their interest in the multicast information that the multicast source S sends to the multicast group G.

After receiving a report message, the DR first checks whether the group address in this message is in the SSM group range and does the following:

- If the group address is in the SSM group range, the DR sends a subscribe message hop by hop toward the multicast source S. All devices along the path from the DR to the source create an (S, G) entry to build an SPT. The SPT is rooted at the multicast source S and has the receivers as its leaves. This SPT is the transmission channel in PIM-SSM.
- If the group address is not in the SSM group range, the receiver-side DR sends a (*, G) join message to the RP. The multicast source registers with the source-side DR.

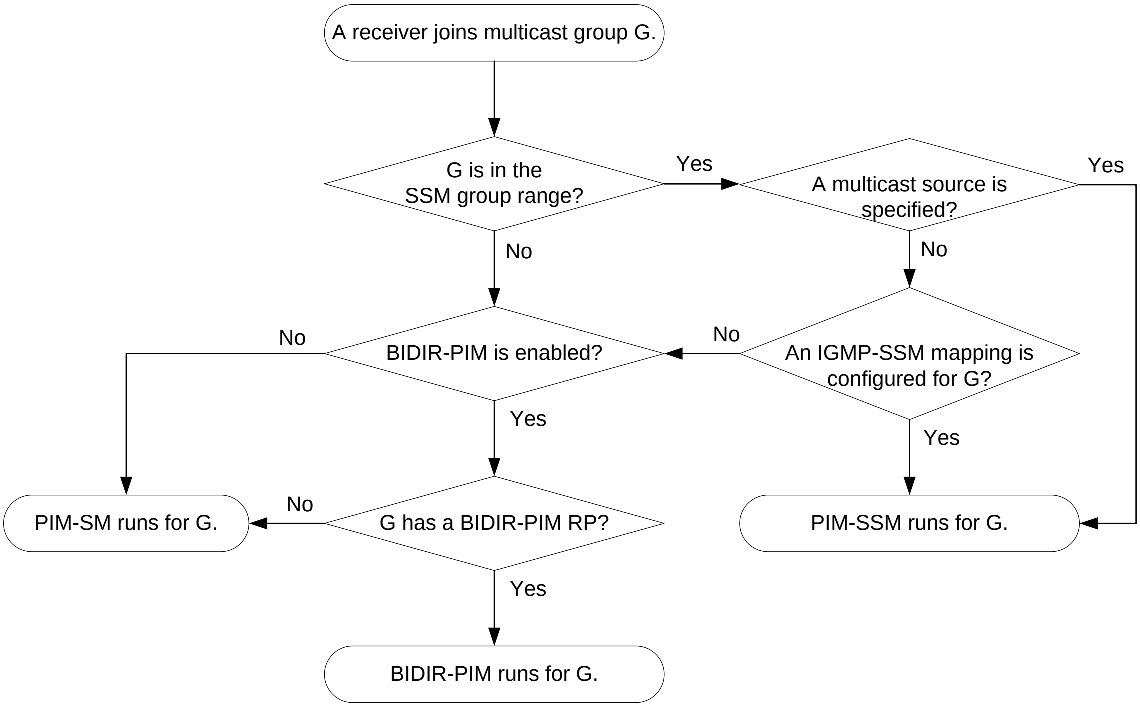
In PIM-SSM, the term "subscribe message" refers to a join message.

Relationship among PIM protocols

In a PIM network, PIM-DM cannot run together with PIM-SM, BIDIR-PIM, or PIM-SSM. However, PIM-SM, BIDIR-PIM, and PIM-SSM can run together. Figure 14 shows how the device selects one protocol from among them for a receiver trying to join a group.

For more information about IGMP SSM mapping, see "Configuring IGMP."

Figure 14 Relationship among PIM protocols


A receiver joins multicast group G.
G is in the SSM group range?
Yes
A multicast source is specified?
Yes
No
No
No
BIDIR-PIM is enabled?
No
An IGMP-SSM mapping is configured for G?
Yes
Yes
PIM-SM runs for G.
No
G has a BIDIR-PIM RP?
PIM-SSM runs for G.
Yes
BIDIR-PIM runs for G.

PIM support for VPNs

To support PIM for VPNs, a multicast device that runs PIM maintains an independent set of PIM neighbor table, multicast routing table, BSR information, and RP-set information for each VPN.

After receiving a multicast data packet, the multicast device checks which VPN the data packet belongs to. Then, the device forwards the packet according to the multicast routing table for that VPN or creates a multicast routing entry for that VPN.

Protocols and standards

- RFC 3973, *Protocol Independent Multicast-Dense Mode (PIM-DM): Protocol Specification(Revised)*
- RFC 4601, *Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification (Revised)*
- RFC 4610, *Anycast-RP Using Protocol Independent Multicast (PIM)*
- RFC 5015, *Bidirectional Protocol Independent Multicast (BIDIR-PIM)*
- RFC 5059, *Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)*
- RFC 4607, *Source-Specific Multicast for IP*
- Draft-ietf-ssm-overview-05, *An Overview of Source-Specific Multicast (SSM)*

Configuring PIM

Restrictions and guidelines: PIM configuration

All the interfaces on a device must operate in the same PIM mode on the public network or the same VPN instance.

When both PIM-SM and BIDIR-PIM run on the PIM network, do not use the same RP to provide services for PIM-SM and BIDIR-PIM. Otherwise, exceptions might occur to the PIM routing table.

Configuring PIM-DM

PIM-DM tasks at a glance

To configure PIM-DM, perform the following tasks:

1. Enabling PIM-DM
2. (Optional.) Configuring the state refresh feature
3. (Optional.) Setting the PIM-DM graft retry timer
4. (Optional.) Configuring common PIM timers

Prerequisites for PIM-DM

Before you configure PIM-DM, complete the following tasks:

- Configure a unicast routing protocol so that all devices in the domain can interoperate at the network layer.
- Enable IP multicast routing.

Enabling PIM-DM

About this task

With PIM-DM enabled on interfaces, devices can establish PIM neighbor relationship and process PIM messages from their PIM neighbors.

Restrictions and guidelines

As a best practice, enable PIM-DM on all non-border interfaces of the devices when you deploy a PIM-DM domain.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable IP multicast routing and enter MRIB view.

   ```
   multicast routing [ vpn-instance vpn-instance-name ]
   ```

   By default, IP multicast routing is disabled.

   For more information about this command, see *IP Multicast Command Reference*.

3. Return to system view.

   ```
   quit
   ```

4. Enter interface view.

   interface *interface-type interface-number*

5. Enable PIM-DM.

   pim dm

   By default, PIM-DM is disabled.

Configuring the state refresh feature

About this task

- **State refresh capability**—Enables the PIM device directly connected to the source to periodically send state refresh messages. It also enables other PIM devices to refresh pruned state timers after receiving the state refresh messages. It prevents the pruned interfaces from resuming multicast forwarding.
- **State refresh interval**—Determines the interval at which a device sends state refresh messages.
- **Wait time before accepting a new state refresh message**—A device might receive duplicate state refresh messages within a short time. To prevent this situation, you can configure the time that the device must wait to accept a new state refresh message. If the device receives a new state refresh message before the timer expires, it discards the message. If the device receives a new state refresh message after the timer expires, it accepts the message, refreshes its own PIM-DM state, and resets the waiting timer.
- **TTL value of state refresh messages**—The TTL value of a state refresh message decrements by 1 whenever it passes a device before it is forwarded to the downstream node. The state refresh message is not forwarded when the TTL value comes down to 0. A state refresh message with a large TTL value might cycle on a small network.

  To effectively control the propagation scope of state refresh messages, configure an appropriate TTL value based on the network size on the device directly connected with the multicast source.

Restrictions and guidelines

Perform this task on all devices in the PIM-DM domain.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable the state refresh feature.

   pim state-refresh-capable

   By default, the state refresh feature is enabled.

4. Return to system view.

   quit

5. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

6. Set the state refresh interval.

   state-refresh-interval *interval*

   The default setting is 60 seconds.

   Perform this task on the device directly connected to the multicast source.

7. Set the amount of time that the device must wait to accept a new state refresh message.

```
state-refresh-rate-limit time
```

The default setting is 30 seconds.

8. Set the TTL value of state refresh messages.

```
state-refresh-ttl ttl-value
```

The default setting is 255.

Setting the PIM-DM graft retry timer

About this task

Perform this task to adjust the interval at which the device retransmits a graft message if it does not receive a graft-ack message from the upstream device.

Procedure

1. Enter system view.

```
system-view
```

2. Enter interface view.

```
interface interface-type interface-number
```

3. Set the graft retry timer.

```
pim timer graft-retry interval
```

The default setting is 3 seconds.

Configuring PIM-SM

PIM-SM tasks at a glance

To configure PIM-SM, perform the following tasks:

1. Enabling PIM-SM
2. Configuring static RPs

   As a best practice, configure a static RP when only one dynamic RP exists in the network.
3. Configuring dynamic RPs
   - Configuring C-RPs
   - Configuring C-BSRs
   - (Optional.) Configuring a PIM domain border
   - (Optional.) Disabling BSM semantic fragmentation
   - (Optional.) Disabling the device from forwarding BSMs out of their incoming interfaces

   As a best practice, configure dynamic RPs when multiple PIM devices exist in the network.

   You can configure static RPs, dynamic RPs, or both.
4. (Optional.) Enabling Auto-RP listening
5. (Optional.) Configuring Anycast RP
6. (Optional.) Configuring multicast source registration
7. (Optional.) Configuring the switchover to SPT
8. (Optional.) Configuring common PIM features

Prerequisites for PIM-SM

Before you configure PIM-SM, complete the following tasks:

- Configure a unicast routing protocol so that all devices in the domain can interoperate at the network layer.
- Enable IP multicast routing.

Enabling PIM-SM

About this task

With PIM-SM enabled on interfaces, devices can establish PIM neighbor relationship and process PIM messages from their PIM neighbors.

Restrictions and guidelines

As a best practice, enable PIM-SM on all non-border interfaces of devices when you deploy a PIM-SM domain.

Procedure

1. Enter system view.

   `system-view`

2. Enable IP multicast routing and enter MRIB view.

   `multicast routing` [**`vpn-instance`** *`vpn-instance-name`*]

   By default, IP multicast routing is disabled.

   For more information about this command, see *IP Multicast Command Reference*.

3. Return to system view.

   `quit`

4. Enter interface view.

   `interface` *`interface-type interface-number`*

5. Enable PIM-SM.

   `pim sm`

   By default, PIM-SM is disabled.

Configuring static RPs

About this task

If only one dynamic RP exists on a network, you can configure a static RP to avoid communication interruption caused by single-point failures. The static RP can also avoid waste of bandwidth because of frequent message exchanges between C-RPs and the BSR.

Restrictions and guidelines

In a PIM-SM domain, you can configure the same static RP for different multicast groups by using the same RP address but different ACLs.

You do not need to enable PIM for an interface to be configured as a static RP.

If you configure multiple static RPs for a multicast group, only the static RP with the highest IP address takes effect.

The static RP configuration must be the same on all devices in the PIM-SM domain.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a static RP.

   static-rp *rp-address* [*ipv4-acl-number* | **preferred**] *

Configuring C-RPs

About this task

To avoid C-RP spoofing, configure a C-RP policy to filter C-RP advertisement messages by using an ACL that specifies the packet source address range and multicast group addresses.

Restrictions and guidelines

Configure C-RPs on devices that reside in the backbone network.

Because the RP and other devices exchange a large amount of information in the PIM-SM domain, reserve a large bandwidth between C-RPs and other devices.

You must configure the same C-RP policy on all C-BSRs in the PIM-SM domain because every C-BSR might become the BSR.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a C-RP.

   c-rp *ip-address* [**advertisement-interval** *adv-interval* | **group-policy** *ipv4-acl-number* | **holdtime** *hold-time* | **priority** *priority*] *

4. (Optional.) Configure a C-RP policy.

   crp-policy *ipv4-acl-number*

   By default, no C-RP policies are configured. All C-RP advertisement messages are regarded as legal.

Configuring C-BSRs

About this task

You must configure C-BSRs when you configure dynamic RP election.

To prevent a legal BSR from being replaced by a malicious host, configure a BSR policy to filter BSR messages by using an ACL that specifies the legal BSR addresses. It is used to prevent the legal BSR from being replaced by a malicious host.

Restrictions and guidelines

Configure C-BSRs on devices that reside in the backbone network.

Because the BSR and other devices exchange a large amount of information in the PIM-SM domain, reserve a large bandwidth between C-BSRs and other devices.

The C-BSR configuration on the devices in the PIM-SM domain must be the same.

For a successful RPF check, configure static multicast routes to ensure that the next hop to a C-BSR is a tunnel interface when C-BSRs connect to other PIM devices through tunnels. For more information about static multicast routes, see "Configuring multicast routing and forwarding."

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a C-BSR.

   c-bsr *ip-address* [**scope** *group-address* { *mask-length* | *mask* }] [**hash-length** *hash-length* | **priority** *priority*] *

4. (Optional.) Configure a BSR policy.

   bsr-policy *ipv4-acl-number*

   By default, no BSR policies are configured. All bootstrap messages are regarded as legal.

Configuring a PIM domain border

About this task

A PIM domain border determines the transmission boundary of bootstrap messages. Bootstrap messages cannot cross the domain border in either direction. A number of PIM domain border interfaces partition a network into different PIM-SM domains.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a PIM domain border.

   pim bsr-boundary

   By default, an interface is not a PIM domain border.

Disabling BSM semantic fragmentation

About this task

BSM semantic fragmentation enables a BSR to split a BSM into multiple BSM fragments (BSMFs) if the BSM exceeds the MTU. In this way, a non-BSR device can update the RP-set information for a group range after receiving all BSMFs for the range. The loss of one BSMF only affects the RP-set information of the group ranges that the fragment contains.

Restrictions and guidelines

If a device does not support BSM semantic fragmentation, it regards a BSMF as a BSM and updates the RP-set information each time it receives a BSMF. It learns only part of the RP-set information, which further affects the RP election. Therefore, if such a device exists in the PIM-SM domain, you must disable BSM semantic fragmentation on all C-BSRs.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Disable BSM semantic fragmentation.

   undo bsm-fragment enable

   By default, BSM semantic fragmentation is enabled.

Disabling the device from forwarding BSMs out of their incoming interfaces

About this task

By default, the device forwards BSMs out of their incoming interfaces. This default setting ensures that all devices on the subnet can receive BSMs even when they have inconsistent routing information. However, this default setting results in duplicated multicast traffic. If you are sure that all the devices on the subnet have consistent routing information, you can disable the device from forwarding BSMs out of their incoming interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Disable the device from forwarding BSMs out of their incoming interfaces.

   undo bsm-reflection enable

   By default, the device forwards BSMs out of their incoming interfaces.

Enabling Auto-RP listening

About this task

This feature enables the device to receive Auto-RP announcement and discovery messages and learn RP information. The destination IP addresses for Auto-RP announcement and discovery messages are 224.0.1.39 and 224.0.1.40, respectively.

Restrictions and guidelines

After this feature is enabled, the device can receive and forward Auto-RP announcement and discovery messages, but it cannot send these messages unsolicitedly.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Enable Auto-RP listening.

   auto-rp enable

   By default, Auto-RP listening is disabled.

Configuring Anycast RP

Restrictions and guidelines

To prevent the other Anycast RP member devices from discarding the BSM sent by the BSR, make sure the Anycast RP address is different from the BSR address.

As a best practice to ensure network performance, configure a maximum of 16 Anycast RP members for an Anycast RP set.

As a best practice, configure the IP address of a loopback interface as the RP member address.

If you configure IP addresses of multiple interfaces on the same device as RP member addresses, the lowest IP address takes effect. The rest of the interface addresses become backup RP member addresses.

An Anycast RP address must be a host address with subnet mask 255.255.255.255.

You must add the device where the Anycast RP resides as an RP member to the Anycast RP set.

Prerequisites

You must configure a static RP or C-RPs in the PIM-SM domain before you configure the Anycast RP. Use the address of the static RP or the dynamically elected RP as the Anycast RP address.

Procedure

1. Enter system view.

   `system-view`

2. Enter PIM view.

   `pim` [**`vpn-instance`** *`vpn-instance-name`*]

3. Configure Anycast RP.

   `anycast-rp` *`anycast-rp-address member-rp-address`*

   By default, Anycast RP is not configured.

   You can repeat this command to add multiple RP member addresses to the Anycast RP set.

Configuring multicast source registration

About this task

- **PIM register policy**—A PIM register policy enables an RP to filter register messages by using an ACL that specifies the multicast sources and groups. The policy limits the multicast groups to which the RP is designated. If a register message is denied by the ACL or does not match the ACL, the RP discards the register message and sends a register-stop message to the source-side DR. The registration process stops.
- **Checksum computing method for register messages**—For information integrity of a register message, you can configure the device to calculate the checksum based on the entire register message. If a device cannot calculate the checksum based on the entire register message, you can configure the device to calculate the checksum based on the register message header.
- **Register suppression time—**The source-side DR stops sending register messages encapsulated with multicast data and starts a register-stop timer upon receiving a register-stop message from the RP. Before the register-stop timer expires, the DR sends a null register message (a register message without encapsulated multicast data) to the RP and starts a register probe timer. If the DR receives a register-stop message before the register probe timer expires, it resets its register-stop timer. Otherwise, the DR starts sending register messages with encapsulated data again.

The register-stop timer is set to a random value chosen uniformly from (0.5 × register_suppression_time minus register_probe_time) to (1.5 × register_suppression_time minus register_probe_time). The register_probe_time is fixed to 5 seconds. You can adjust the register suppression time.

Restrictions and guidelines

On all C-RP devices, configure a PIM register policy and the checksum computing method for register messages.

On all devices that might become the source-side DR, set the register suppression time.

Procedure

1. Enter system view.

   `system-view`

2. Enter PIM view.

   `pim` [**`vpn-instance`** *`vpn-instance-name`*]

3. Configure a PIM register policy.

   `register-policy` *`ipv4-acl-number`*

   By default, no PIM register policy is configured. All PIM register messages are regarded as legal.

4. Configure the device to calculate the checksum based on the entire register message.

   `register-whole-checksum`

   By default, the device calculates the checksum based on the header of a register message.

5. Set the register suppression time.

   `register-suppression-timeout` *`interval`*

   The default setting is 60 seconds.

Configuring the switchover to SPT

About this task

Both the receiver-side DR and RP can monitor the traffic rate of passing-by multicast packets and thus trigger a switchover from RPT to SPT. The monitor function is not available on switches.

Restrictions and guidelines

Some devices cannot encapsulate multicast data in register messages destined to the RP. As a best practice to avoid multicast data forwarding failures, do not disable the switchover to SPT on C-RPs that might become the RP.

Procedure

1. Enter system view.

   `system-view`

2. Enter PIM view.

   `pim` [**`vpn-instance`** *`vpn-instance-name`*]

3. Configure RPT to SPT switchover.

   `spt-switch-threshold` { *`traffic-rate`* | **`immediacy`** | **`infinity`** } [**`group-policy`** *`ipv4-acl-number`*]

   By default, the first multicast data packet triggers the RPT to SPT switchover.

   Support for the *`traffic-rate`* argument depends on the device model. For more information, see the command reference.

Configuring BIDIR-PIM

BIDIR-PIM tasks at a glance

To configure BIDIR-PIM, perform the following tasks:

1. Enabling BIDIR-PIM
2. Configuring static RPs

   As a best practice, configure a static RP when only one dynamic RP exists in the network.
3. Configuring dynamic RPs
   - Configuring C-RPs
   - Configuring C-BSRs
   - (Optional.) Configuring a PIM domain border
   - (Optional.) Disabling BSM semantic fragmentation
   - (Optional.) Disabling the device from forwarding BSMs out of their incoming interfaces

   As a best practice, configure dynamic RPs when multiple PIM devices exist in the network.

   You can configure static RPs, dynamic RPs, or both.
4. (Optional.) Setting the maximum number of BIDIR-PIM RPs
5. (Optional.) Enabling Auto-RP listening
6. (Optional.) Configuring common PIM features

Prerequisites for BIDIR-PIM

Before you configure BIDIR-PIM, complete the following tasks:

- Configure a unicast routing protocol so that all devices in the domain can interoperate at the network layer.
- Enable PIM-SM.

Enabling BIDIR-PIM

Restrictions and guidelines

As a best practice, enable PIM-SM on all non-border interfaces of devices when you deploy a BIDIR-PIM domain.

Procedure

1. Enter system view.

   `system-view`
2. Enable IP multicast routing and enter MRIB view.

   `multicast routing` [**`vpn-instance`** *`vpn-instance-name`*]

   By default, IP multicast routing is disabled.

   For more information about this command, see *IP Multicast Command Reference*.
3. Return to system view.

   `quit`
4. Enter interface view.

   `interface` *`interface-type interface-number`*
5. Enable PIM-SM.

```
pim sm
```

By default, PIM-SM is disabled.

6. Return to system view.

   ```
   quit
   ```

7. Enter PIM view.

   ```
   pim [ vpn-instance vpn-instance-name ]
   ```

8. Enable BIDIR-PIM.

   ```
   bidir-pim enable
   ```

   By default, BIDIR-PIM is disabled.

Configuring static RPs

About this task

If only one dynamic RP exists on a network, you can configure a static RP to avoid communication interruption caused by single-point failures. The static RP can also avoid bandwidth waste caused by frequent message exchanges between C-RPs and the BSR.

Restrictions and guidelines

The static RP configuration must be the same on all devices in the BIDIR-PIM domain.

In a BIDIR-PIM domain, you can configure the same static RP for different multicast groups by using the same RP address but different ACLs.

You do not need to enable PIM for an interface to be configured as a static RP.

If you configure multiple static RPs for a multicast group, only the static RP with the highest IP address takes effect.

You can specify an unused IP address for a static RP. This address must be on the same subnet with the link on which the static RP is configured. For example, if the IP addresses of the interfaces at the two ends of a link are 10.1.1.1/24 and 10.1.1.2/24, you can specify the interface with IP address 10.1.1.100/24 as a static RP. As a result, the link becomes an RPL.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter PIM view.

   ```
   pim [ vpn-instance vpn-instance-name ]
   ```

3. Configure a static RP for BIDIR-PIM.

   ```
   static-rp rp-address bidir [ ipv4-acl-number | preferred ] *
   ```

Configuring C-RPs

About this task

To guard against C-RP spoofing, configure a C-RP policy to filter C-RP advertisement messages by using an ACL that specifies the packet source address range and multicast groups.

Restrictions and guidelines

Configure C-RPs on devices that reside in the backbone network.

Because the RP and other devices exchange a large amount of information in the BIDIR-PIM domain, reserve a large bandwidth between C-RPs and other devices.

You must configure the same C-RP policy on all C-BSRs in the BIDIR-PIM domain because every C-BSR might become the BSR.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a C-RP to provide services for BIDIR-PIM.

   c-rp *ip-address* [**advertisement-interval** *adv-interval* | **group-policy** *ipv4-acl-number* | **holdtime** *hold-time* | **priority** *priority*] * **bidir**

Configuring C-BSRs

About this task

You must configure C-BSRs when you configure dynamic RP election.

To prevent a legal BSR from being replaced by a malicious host, configure a BSR policy to filter BSR messages by using an ACL that specifies the legal BSR addresses.

Restrictions and guidelines

Configure C-BSRs on devices that reside in the backbone device.

Because the BSR and other devices exchange a large amount of information in the BIDIR-PIM domain, reserve a large bandwidth between the C-BSRs and other devices.

The C-BSR configuration on the devices in the BIDIR-PIM domain must be the same.

For C-BSRs interconnected through a GRE tunnel, configure static multicast routes to make sure the next hop to a C-BSR is a tunnel interface. For more information about static multicast routes, see "Configuring multicast routing and forwarding."

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a C-BSR.

   c-bsr *ip-address* [**scope** *group-address* { *mask-length* | *mask* }] [**hash-length** *hash-length* | **priority** *priority*] *

4. (Optional.) Configure a BSR policy.

   bsr-policy *ipv4-acl-number*

   By default, no BSR policy is configured. All bootstrap messages are regarded as legal.

Configuring a PIM domain border

About this task

A PIM domain border determines the transmission boundary of bootstrap messages. Bootstrap messages cannot cross the domain border in either direction. A number of PIM domain border interfaces partition a network into different BIDIR-PIM domains.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure a PIM domain border.

   pim bsr-boundary

   By default, an interface is not a PIM domain border.

Disabling BSM semantic fragmentation

About this task

BSM semantic fragmentation enables a BSR to split a BSM into multiple BSM fragments (BSMFs) if the BSM exceeds the MTU. In this way, a non-BSR device can update the RP-set information for a group range after receiving all BSMFs for the group range. The loss of one BSMF only affects the RP-set information of the group ranges that the fragment contains.

Restrictions and guidelines

If a device does not support BSM semantic fragmentation, it regards a BSMF as a BSM and updates the RP-set information each time it receives a BSMF. It learns only part of the RP-set information, which further affects the RP election. Therefore, if such a device presents in the BIDIR-PIM domain, you must disable BSM semantic fragmentation on all C-BSRs.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Disable BSM semantic fragmentation.

   undo bsm-fragment enable

   By default, BSM semantic fragmentation is enabled.

Disabling the device from forwarding BSMs out of their incoming interfaces

About this task

By default, the device forwards BSMs out of their incoming interfaces. This default setting ensures that all devices on the subnet can receive BSMs even when they have inconsistent routing information. However, this default setting results in duplicated multicast traffic. If you are sure that all the devices on the subnet have consistent routing information, you can disable the device from forwarding BSMs out of their incoming interfaces.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Disable the device from forwarding BSMs out of their incoming interfaces.

undo bsm-reflection enable

By default, the device forwards BSMs out of their incoming interfaces.

Setting the maximum number of BIDIR-PIM RPs

About this task

In a BIDIR-PIM domain, one DF election per RP is implemented on all PIM-enabled interfaces. To avoid unnecessary DF elections, do not configure multiple RPs for BIDIR-PIM.

This configuration sets a limit on the number of BIDIR-PIM RPs. If the number of RPs exceeds the limit, excess RPs do not take effect and can be used only for DF election rather than multicast data forwarding. The system does not delete the excess RPs. They must be deleted manually.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Set the maximum number of BIDIR-PIM RPs.

   bidir-rp-limit *limit*

   By default, the maximum number of BIDIR-PIM RPs is 128.

Enabling Auto-RP listening

About this task

This feature enables the device to receive Auto-RP announcement and discovery messages and learn RP information. The destination IP addresses for Auto-RP announcement and discovery messages are 224.0.1.39 and 224.0.1.40, respectively.

Restrictions and guidelines

After this feature is enabled, the device can receive and forward Auto-RP announcement and discovery messages, but it cannot send these messages unsolicitedly.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Enable Auto-RP listening.

   auto-rp enable

   By default, Auto-RP listening is disabled.

Configuring PIM-SSM

PIM-SSM tasks at a glance

To configure PIM-SSM, perform the following tasks:

1. Enabling PIM-SM

2. (Optional.) Configuring the SSM group range
3. (Optional.) Configuring common PIM features

Prerequisites for PIM-SSM

Before you configure PIM-SSM, complete the following tasks:

- Configure a unicast routing protocol so that all devices in the domain can interoperate at the network layer.
- Enable PIM-SM.
- Enable IGMPv3 on PIM devices that connect to multicast receivers.

Enabling PIM-SM

Restrictions and guidelines

As a best practice, enable PIM-SM on all non-border interfaces of devices when you deploy a PIM-SSM domain.

Procedure

1. Enter system view.

   `system-view`

2. Enable multicast routing and enter MRIB view.

   `multicast routing` [**`vpn-instance`** *`vpn-instance-name`*]

   By default, multicast routing is disabled.

   For more information about this command, see *IP Multicast Command Reference*.

3. Return to system view.

   `quit`

4. Enter interface view.

   `interface` *`interface-type interface-number`*

5. Enable PIM-SM.

   `pim sm`

   By default, PIM-SM is disabled.

Configuring the SSM group range

About this task

When a PIM-SM enabled interface receives a multicast packet, it checks whether the multicast group address of the packet is in the SSM group range. If the multicast group address is in this range, the PIM mode for this packet is PIM-SSM. If the multicast group address is not in this range, the PIM mode is PIM-SM.

Restrictions and guidelines

Configure the same SSM group range on all devices in the entire PIM-SSM domain. Otherwise, multicast information cannot be delivered through the SSM model.

When a member of a multicast group in the SSM group range sends an IGMPv1 or IGMPv2 report message, the device does not trigger a (*, G) join.

Procedure

1. Enter system view.

system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure the SSM group range.

   ssm-policy *ipv4-acl-number*

   By default, the SSM group range is 232.0.0.0/8.

Configuring common PIM features

Common PIM feature tasks at a glance

All the following tasks are optional.

- Configuring a multicast source policy
- Configuring a PIM hello policy
- Configuring PIM hello message options
- Dropping hello messages without the Generation ID option
- Configuring common PIM timers
- Setting the maximum size of a join or prune message
- Enabling PIM prune delay
- Enabling BFD for PIM
- Enabling PIM passive mode
- Enabling PIM NSR
- Enabling NBMA mode for ADVPN tunnel interfaces
- Enabling SNMP notifications for PIM

Configuring a multicast source policy

About this task

This feature enables the device to filter multicast data by using an ACL that specifies the multicast sources and the optional groups. It filters not only data packets but also register messages with multicast data encapsulated. It controls the information available to downstream receivers.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Configure a multicast source policy.

   source-policy *ipv4-acl-number*

   By default, no multicast source policy is configured. The device does not filter multicast data packets.

Configuring a PIM hello policy

About this task

This feature enables the device to filter PIM hello messages by using an ACL that specifies the packet source addresses. It is used to guard against PIM message attacks and to establish correct PIM neighboring relationships.

If hello messages of an existing PIM neighbor are filtered out by the policy, the neighbor is automatically removed when its aging timer expires.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure a PIM hello policy.

   `pim neighbor-policy` *`ipv4-acl-number`*

   By default, no PIM hello policy is configured on an interface. All PIM hello messages are regarded as legal.

Configuring PIM hello message options

About this task

In either a PIM-DM domain or a PIM-SM domain, hello messages exchanged among devices contain the following configurable options:

- **DR_Priority** (for PIM-SM only)—Priority for DR election. The device with the highest priority wins the DR election. You can configure this option for all the devices in a shared-media LAN that directly connects to the multicast source or the receivers.
- **Holdtime**—PIM neighbor lifetime. If a device does not receive a hello message from a neighbor when the neighbor lifetime expires, it regards the neighbor failed or unreachable.
- **LAN_Prune_Delay**—Delay of pruning a downstream interface on a shared-media LAN. This option has the LAN delay field, the override interval field, and the T bit.

  The LAN delay defines the PIM message propagation delay. The override interval defines a period for a device to override a prune message. The T bit specifies the capability of tracking downstream device status on upstream devices.

  On the shared-media LAN, the propagation delay and override interval are used as follows:

  - If a device receives a prune message on its upstream interface, it means that there are downstream devices on the shared-media LAN. If this device still needs to receive multicast data, it must send a join message to override the prune message within the override interval.
  - When a device receives a prune message from its downstream interface, it does not immediately prune this interface. Instead, it starts a timer (the propagation delay plus the override interval). If interface receives a join message before the timer expires, the device does not prune the interface. Otherwise, the device prunes the interface.
- **Neighbor tracking**—If you enable neighbor tracking on an upstream device, this device can track the states of the downstream nodes for which the joined state holdtime timer has not expired. All join messages from downstream devices are accepted.

Restrictions and guidelines

If the propagation delay or override interval on different PIM devices on a shared-media LAN are different, the largest ones apply.

If you want to enable neighbor tracking, you must enable it on all PIM devices on a shared-media LAN. Otherwise, the upstream device cannot track join messages from every downstream devices.

You can configure hello message options for all interfaces in PIM view or for the current interface in interface view. The configuration made in interface view takes priority over the configuration made in PIM view.

Configuring hello message options globally

1. Enter system view.
   system-view
2. Enter PIM view.
   pim [**vpn-instance** *vpn-instance-name*]
3. Set the DR priority.
   hello-option dr-priority *priority*
   The default setting is 1.
4. Set the neighbor lifetime.
   hello-option holdtime *time*
   The default setting is 105 seconds.
5. Set the PIM message propagation delay for a shared-media LAN.
   hello-option lan-delay *delay*
   The default setting is 500 milliseconds.
6. Set the override interval.
   hello-option override-interval *interval*
   The default setting is 2500 milliseconds.
7. Enable neighbor tracking.
   hello-option neighbor-tracking
   By default, neighbor tracking is disabled.

Configuring hello message options on an interface

1. Enter system view.
   system-view
2. Enter interface view.
   interface *interface-type interface-number*
3. Set the DR priority.
   pim hello-option dr-priority *priority*
   The default setting is 1.
4. Set the neighbor lifetime.
   pim hello-option holdtime *time*
   The default setting is 105 seconds.
5. Set the PIM message propagation delay.
   pim hello-option lan-delay *delay*
   The default setting is 500 milliseconds.
6. Set the override interval.
   pim hello-option override-interval *interval*
   The default setting is 2500 milliseconds.
7. Enable neighbor tracking.

```
pim hello-option neighbor-tracking
```

By default, neighbor tracking is disabled.

Dropping hello messages without the Generation ID option

About this task

A device generates a generation ID for hello messages when an interface is enabled with PIM. The generation ID is a random value, but it changes only when the status of the device changes. If a PIM device finds that the generation ID in a hello message from the upstream device has changed, it assumes that the status of the upstream device has changed. In this case, it sends a join message to the upstream device for status update. You can configure an interface to drop hello messages without the generation ID options to promptly know the status of an upstream device.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Enable the interface to drop hello messages without the Generation ID option.

   ```
   pim require-genid
   ```

   By default, an interface accepts hello messages without the Generation ID option.

Configuring common PIM timers

About this task

The following are common PIM timers:

- **Hello interval**—Interval at which a PIM device sends hello messages to discover PIM neighbors, and maintain PIM neighbor relationship.
- **Triggered hello delay**—Maximum delay for sending a hello message to avoid collisions caused by simultaneous hello messages. After receiving a hello message, a PIM device waits for a random time before sending a hello message. This random time is in the range of 0 to the triggered hello delay.
- **Join/Prune interval**—Interval at which a PIM device sends join/prune messages to its upstream devices for state update.
- **Joined/Pruned state holdtime**—Time for which a PIM device keeps the joined or pruned state for the downstream interfaces. This joined or pruned state holdtime is specified in a join/prune message.
- **Multicast source lifetime**—Lifetime that a PIM device maintains for a multicast source. If a device does not receive subsequent multicast data from the multicast source S when the timer expires, it deletes the (S, G) entry for the multicast source.

Restrictions and guidelines

To prevent upstream neighbors from aging out, set the join/prune interval to be less than the join/pruned state holdtime.

You can configure common PIM timers for all interfaces in PIM view or for the current interface in interface view. The configuration made in interface view takes priority over the configuration made in PIM view.

As a best practice, use the defaults for a network without special requirements.

Configuring common PIM timers globally

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Set the hello interval.

   timer hello *interval*

   The default setting is 30 seconds.

4. Set the join/prune interval.

   timer join-prune *interval*

   The default setting is 60 seconds.

   This configuration takes effect after the current interval ends.

5. Set the joined or pruned state holdtime.

   holdtime join-prune *time*

   The default setting is 210 seconds.

6. Set the multicast source lifetime.

   source-lifetime *time*

   The default setting is 210 seconds.

Configuring common PIM timers on an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the hello interval.

   pim timer hello *interval*

   The default setting is 30 seconds.

4. Set the triggered hello delay.

   pim triggered-hello-delay *delay*

   The default setting is 5 seconds.

5. Set the join/prune interval.

   pim timer join-prune *interval*

   The default setting is 60 seconds.

   This configuration takes effect after the current interval ends.

6. Set the joined or pruned state holdtime.

   pim holdtime join-prune *time*

   The default setting is 210 seconds.

Setting the maximum size of a join or prune message

About this task

The loss of an oversized join or prune message might result in loss of massive information. You can set a small value for the size of a join or prune message to reduce the impact.

Procedure

1. Enter system view.

   system-view

2. Enter PIM view.

   pim [**vpn-instance** *vpn-instance-name*]

3. Set the maximum size of a join or prune message.

   jp-pkt-size *size*

   The default setting is 1200 bytes.

Enabling PIM prune delay

About this task

By default, an interface determines whether to starts the prune delay timer when it receives a prune message based on the number of PIM neighbors it. An interface starts the prune delay timer only when it has more than one PIM neighbor. When the prune delay timer expires, the device remove the receiving interface from the output interface list of the (S, G) entry.

In a DRNI-capable Layer 3 multicast network, the IP addresses of the two DR interfaces on DR devices are the same, the upstream device will consider the DR devices as one PIM neighbor and will not start the prune delay timer. When one DR device sends a prune message to the upstream device, the upstream device immediately remote the DR interface from the output interface list of the (S, G) entry. If the other DR device has receivers connected and the network is complex, the upstream device need to wait a long time to receive prune messages from the DR device. This situation causes multicast traffic interruption for a long time.

This feature allows the device to enable PIM prune delay and start the prune delay timer regardless of the number of PIM neighbors on an interface. The value of the timer is the sum of the override interval (configured by using the **pim hello-option override-interval** command) and the PIM message propagation delay (configured by using the **pim hello-option lan-delay** command).

Restrictions and guidelines

This feature takes effect only if PIM-DM or PIM-SM is enabled.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable BFD for PIM.

   pim prune-pending

   By default, BFD is disabled for PIM.

Enabling BFD for PIM

About this task

If a DR on a shared-media network fails, a new DR election process does not start until the DR ages out. In addition, it might take a long period of time before other devices detect the link failures and trigger a new DR election. To start a new DR election process immediately after the original DR fails, enable BFD for PIM to detect link failures among PIM neighbors.

You must enable BFD for PIM on all PIM devices on a shared-media network. For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable BFD for PIM.

   `pim bfd enable`

   By default, BFD is disabled for PIM.

Enabling PIM passive mode

About this task

To guard against PIM hello spoofing, you can enable PIM passive mode on a receiver-side interface. The PIM passive interface cannot receive or forward PIM protocol messages (excluding register, register-stop, and C-RP-Adv messages), and it acts as the DR on the subnet. In BIDIR-PIM, it also acts as the DF.

Restrictions and guidelines

To avoid duplicate multicast data transmission and flow loop, do not enable this feature on a shared-media LAN with multiple PIM devices.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Enable PIM passive mode on the interface.

   `pim passive`

   By default, PIM passive mode is disabled on an interface.

Enabling PIM NSR

About this task

This feature enables PIM to back up protocol state information and data, including PIM neighbor information and routes, from the active process to the standby process. The standby process immediately takes over when the active process fails. Use this feature to avoid route flapping and forwarding interruption for PIM when an active/standby switchover occurs.

Procedure

1. Enter system view.

   `system-view`

2. Enable PIM NSR.

   `pim non-stop-routing`

   By default, PIM NSR is disabled.

Enabling NBMA mode for ADVPN tunnel interfaces

About this task

This feature allows ADVPN tunnel interfaces to forward multicast data only to target spokes and hubs. For more information about ADVPN, see *VPN Configuration Guide*.

Restrictions and guidelines

This feature is not available for PIM-DM.

This feature takes effect only when PIM-SM is enabled on the ADVPN tunnel interface.

In a BIDIR-PIM domain, make sure RPs do not reside on ADVPN tunnel interfaces or on the subnet where ADVPN tunnel interfaces are located.

Do not configure IGMP features on ADVPN tunnel interfaces that are enabled with NBMA mode.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable NBMA mode.

   pim nbma-mode

   By default, NBMA mode is disabled.

   This command is applicable only to ADVPN tunnel interfaces.

Enabling SNMP notifications for PIM

About this task

To report critical PIM events to an NMS, enable SNMP notifications for PIM. For PIM event notifications to be sent correctly, you must also configure SNMP as described in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable SNMP notifications for PIM.

   snmp-agent trap enable pim [**candidate-bsr-win-election** | **elected-bsr-lost-election** | **neighbor-loss**] *

   By default, SNMP notifications for PIM are enabled.

Display and maintenance commands for PIM

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display register-tunnel interface information.	**display interface** [**register-tunnel** [*interface-number*]] [**brief** [**description** | **down**]]

Task	Command
Display BSR information in the PIM-SM domain.	**display pim** [**vpn-instance** *vpn-instance-name*] **bsr-info**
Display information about the routes used by PIM.	**display pim** [**vpn-instance** *vpn-instance-name*] **claimed-route** [*source-address*]
Display C-RP information in the PIM-SM domain.	**display pim** [**vpn-instance** *vpn-instance-name*] **c-rp** [**local**]
Display DF information in the BIDIR-PIM domain.	**display pim** [**vpn-instance** *vpn-instance-name*] **df-info** [*rp-address*]
Display PIM information on an interface.	**display pim** [**vpn-instance** *vpn-instance-name*] **interface** [*interface-type interface-number*] [**verbose**]
Display PIM neighbor information.	**display pim** [**vpn-instance** *vpn-instance-name*] **neighbor** [*neighbor-address* | **interface** *interface-type interface-number* | **verbose**] *
Display PIM routing entries.	**display pim** [**vpn-instance** *vpn-instance-name*] **routing-table** [*group-address* [**mask** { *mask-length* | *mask* }] | *source-address* [**mask** { *mask-length* | *mask* }] | **flags** *flag-value* | **fsm** | **incoming-interface** *interface-type interface-number* | **mode** *mode-type* | **outgoing-interface** { **exclude** | **include** | **match** } *interface-type interface-number* | **extranet** { **source-vpn-instance** *source-vpn-instance-name* | **source-public-instance** | **receive-vpn-instance** *receive-vpn-instance-name* | **receive-public-instance** }] *
Display RP information in the PIM-SM domain.	**display pim** [**vpn-instance** *vpn-instance-name*] **rp-info** [*group-address*]
Display statistics for PIM packets.	**display pim statistics**
Display remote end information maintained by PIM for ADVPN tunnel interfaces.	**display pim** [**vpn-instance** *vpn-instance-name*] **nbma-link** [**interface** { *interface-type interface-number* }]

Troubleshooting PIM

A multicast distribution tree cannot be correctly built

Symptom

No multicast forwarding entries are established on the devices (including devices directly connected with multicast sources or receivers) in a PIM network. This means that a multicast distribution tree cannot be built correctly.

Solution

To resolve the problem:

1. Use **display ip routing-table** to verify that a unicast route to the multicast source or the RP is available.
2. Use **display pim interface** to verify PIM information on each interface, especially on the RPF interface. If PIM is not enabled on the interfaces, use **pim dm** or **pim sm** to enable PIM-DM or PIM-SM for the interfaces.
3. Use **display pim neighbor** to verify that the RPF neighbor is a PIM neighbor.
4. Verify that PIM and IGMP are enabled on the interfaces that directly connect to the multicast sources or the receivers.
5. Use **display pim interface verbose** to verify that the same PIM mode is enabled on the RPF interface on a device and the connected interface of the device's RPF neighbor.
6. Use **display current-configuration** to verify that the same PIM mode is enabled on all devices. For PIM-SM, verify that the BSR and C-RPs are correctly configured.
7. If the problem persists, contact NSFOCUS Support.

Multicast data is abnormally terminated on an intermediate device

Symptom

An intermediate device can receive multicast data successfully, but the data cannot reach the last-hop device. An interface on the intermediate device receives multicast data but does not create an (S, G) entry in the PIM routing table.

Solution

To resolve the problem:

1. Use **display current-configuration** to verify the multicast forwarding boundary settings. Use **multicast boundary** to change the multicast forwarding boundary settings to make the multicast packet able to cross the boundary.
2. Use **display current-configuration** to verify the multicast source policy. Change the ACL rule defined in the **source-policy** command so that the source/group address of the multicast data can pass ACL filtering.
3. If the problem persists, contact NSFOCUS Support.

An RP cannot join an SPT in PIM-SM

Symptom

An RPT cannot be correctly built, or an RP cannot join the SPT toward the multicast source.

Solution

To resolve the problem:

1. Use **display ip routing-table** to verify that a unicast route to the RP is available on each device.
2. Use **display pim rp-info** to verify that the dynamic RP information is consistent on all devices.
3. Use **display pim rp-info** to verify that the same static RPs are configured on all devices on the network.
4. If the problem persists, contact NSFOCUS Support.

An RPT cannot be built or multicast source registration fails in PIM-SM

Symptom

The C-RPs cannot unicast advertisement messages to the BSR. The BSR does not advertise BSMs containing C-RP information and has no unicast route to any C-RP. An RPT cannot be correctly established, or the source-side DR cannot register the multicast source with the RP.

Solution

To resolve the problem:

1. Use **display ip routing-table** on each device to view routing table information. Verify that unicast routes to the C-RPs and the BSR are available on each device and that a route is available between each C-RP and the BSR.
2. Use **display pim bsr-info** to verify that the BSR information exists on each device.
3. Use **display pim rp-info** to verify that the RP information is correct on each device.
4. Use **display pim neighbor** to verify that PIM neighboring relationship has been correctly established among the devices.
5. If the problem persists, contact NSFOCUS Support.

Contents

Configuring IPv6 multicast routing and forwarding

About IPv6 multicast routing and forwarding

Each IPv6 multicast routing protocol has its own routing table. Multicast routing information in routing entries generated by the IPv6 multicast routing protocols are summarized in a set of (S, G) and (*, G) entries. All the (S, G) and (*, G) entries form a general IPv6 multicast routing table. The optimal IPv6 multicast routing entries in the general IPv6 multicast routing table are added to the IPv6 multicast forwarding table to guide IPv6 multicast data forwarding.

RPF check mechanism

An IPv6 multicast routing protocol uses the reverse path forwarding (RPF) check mechanism to ensure IPv6 multicast data delivery along the correct path and to avoid data loops.

RPF check process

An IPv6 multicast device performs the RPF check on an IPv6 multicast packet as follows:

1. Chooses an optimal route back to the packet source separately from the IPv6 unicast and IPv6 MBGP routing tables.

   In RPF check, the "packet source" means difference things in difference situations:

   - For a packet that travels along the SPT, the packet source is the IPv6 multicast source.
   - For a packet that travels along the RPT, the packet source is the RP.
   - For a bootstrap message originated from the BSR, the packet source is the BSR.

2. Selects one of the optimal routes as the RPF route as follows:

   - If the device uses the longest prefix match principle, the route with a higher prefix length becomes the RPF route. If the routes have the same prefix length, the route with a higher route preference becomes the RPF route. If the routes have the same route preference, the IPv6 MBGP route becomes the RPF route. If equal cost routes exist, the route with the highest next hop IPv6 address becomes the RPF route.

     For more information about the route preference, see *Layer 3—IP Routing Configuration Guide*.

   - If the device does not use the longest prefix match principle, the route with a higher route preference becomes the RPF route. If the routes have the same route preference, the IPv6 MBGP route becomes the RPF route. If equal cost routes exist, the route with the highest next hop IPv6 address becomes the RPF route.

   In the RPF route, the outgoing interface is the RPF interface and the next hop is the RPF neighbor.

3. Determines whether the packet arrived at the RPF interface.

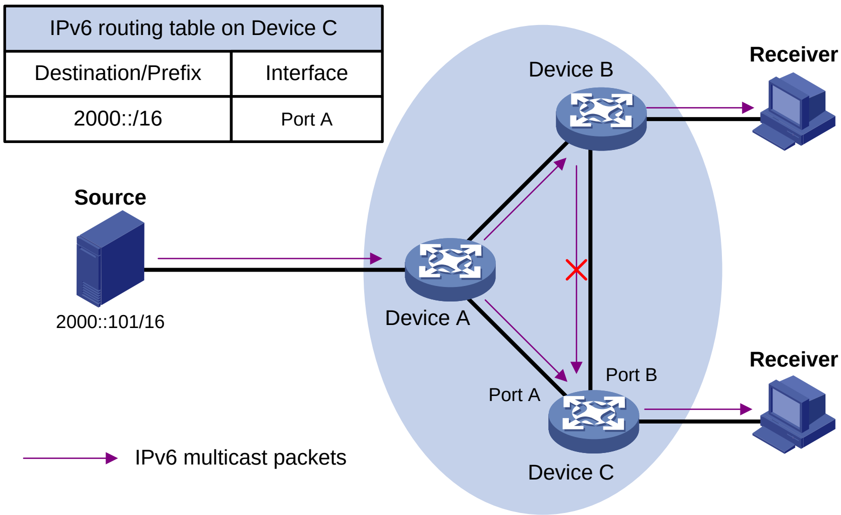
   - If the packet arrived at the RPF interface, the RPF check succeeds and the packet is forwarded.
   - If the packet arrived at the non-RPF interface, the RPF check fails and the packet is discarded.

RPF check implementation in IPv6 multicast

Implementing an RPF check on each received IPv6 multicast packet would heavily burden the device. The use of an IPv6 multicast forwarding table is the solution to this issue. When the device creates an IPv6 multicast forwarding entry for an IPv6 (S, G) packet, it sets the RPF interface of the packet as the incoming interface of the (S, G) entry. After the device receives another (S, G) packet, it looks up its IPv6 multicast forwarding table for a matching (S, G) entry.

- If no match is found, the device first determines the RPF route back to the packet source. Then, it creates a forwarding entry with the RPF interface as the incoming interface and performs one of the following tasks:
    - If the receiving interface is the RPF interface, the RPF check succeeds and the device forwards the packet out of all outgoing interfaces.
    - If the receiving interface is not the RPF interface, the RPF check fails and the device discards the packet.
- If a match is found and the matching forwarding entry contains the receiving interface, the device forwards the packet out of all outgoing interfaces.
- If a match is found but the matching forwarding entry does not contain the receiving interface, the device determines the RPF route back to the packet source. Then, the device performs one of the following tasks:
    - If the RPF interface is the incoming interface, it means that the forwarding entry is correct but the packet traveled along a wrong path. The packet fails the RPF check, and the device discards the packet.
    - If the RPF interface is not the incoming interface, it means that the forwarding entry has expired. The device replaces the incoming interface with the RPF interface and matches the receiving interface against the RPF interface. If the receiving interface is the RPF interface, the device forwards the packet out of all outgoing interfaces. Otherwise, it discards the packet.

Figure 1 RPF check process



IPv6 routing table on Device C	
Destination/Prefix	Interface
2000::/16	Port A

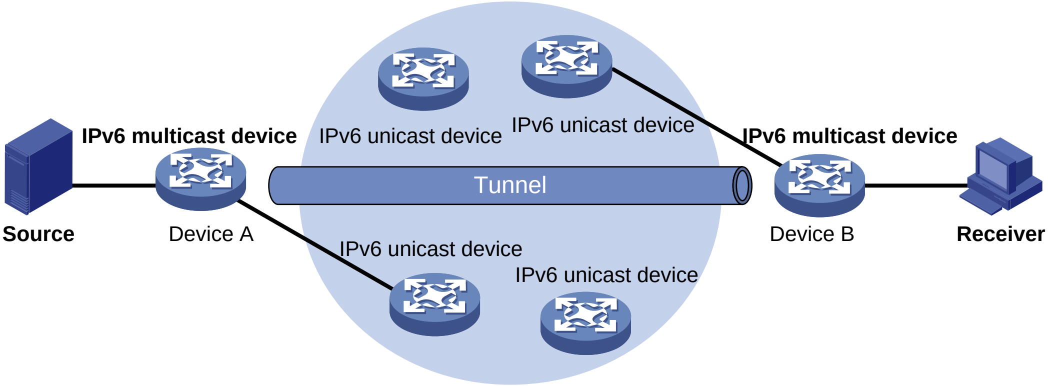
As shown in Figure 1, assume that IPv6 unicast routes are available on the network. IPv6 MBGP is not configured. IPv6 multicast packets travel along the SPT from the multicast source to the receivers. The IPv6 multicast forwarding table on Device C contains the (S, G) entry, with Port A as the RPF interface.

- If an IPv6 multicast packet arrives at Device C on Port A, the receiving interface is the incoming interface of the (S, G) entry. Device C forwards the packet out of all outgoing interfaces.
- If an IPv6 multicast packet arrives at Device C on Port B, the receiving interface is not the incoming interface of the (S, G) entry. Device C searches its IPv6 unicast routing table and finds that the outgoing interface to the source (the RPF interface) is Port A. This means that the (S, G) entry is correct but the packet traveled along a wrong path. The packet fails the RPF check, and Device C discards the packet.

IPv6 multicast forwarding across IPv6 unicast subnets

Devices forward the IPv6 multicast data from an IPv6 multicast source hop by hop along the forwarding tree, but some devices might not support IPv6 multicast protocols in a network. When the IPv6 multicast data is forwarded to a device that does not support IPv6 multicast, the forwarding path is blocked. In this case, you can enable IPv6 multicast data forwarding across the IPv6 unicast subnets by establishing a tunnel between the devices at both ends of the IPv6 unicast subnets.

Figure 2 IPv6 multicast data transmission through a tunnel



As shown in Figure 2, a tunnel is established between Device A and Device B. Device A encapsulates the IPv6 multicast data in unicast IPv6 packets, and forwards them to Device B across the tunnel through unicast devices. Then, Device B strips off the unicast IPv6 header and continues to forward the IPv6 multicast data down toward the receivers.

IPv6 multicast routing and forwarding tasks at a glance

To configure IPv6 multicast routing and forwarding, perform the following tasks:

1. Enabling IPv6 multicast routing
2. (Optional.) Specifying the longest prefix match principle
3. (Optional.) Configuring IPv6 multicast load splitting
4. (Optional.) Configuring an IPv6 multicast forwarding boundary
5. (Optional.) Setting the maximum number of cached unknown IPv6 multicast packets
6. (Optional.) Configuring an IPv6 MVPN extranet RPF selection policy

Prerequisites for IPv6 multicast routing and forwarding

Before you configure multicast routing and forwarding, configure an IPv6 unicast routing protocol so that all devices in the domain can interoperate at the network layer.

Enabling IPv6 multicast routing

About this task

Enable IPv6 multicast routing before you configure any Layer 3 IPv6 multicast functionality in the public network or VPN instance.

Procedure

1. Enter system view.

   system-view

2. Enable IPv6 multicast routing and enter IPv6 MRIB view.

   ipv6 multicast routing [**vpn-instance** *vpn-instance-name*]

   By default, IPv6 multicast routing is disabled.

Specifying the longest prefix match principle

About this task

You can enable the device to use the longest prefix match principle for RPF route selection. For more information about RPF route selection, see "RPF check process."

Procedure

1. Enter system view.

   system-view

2. Enter IPv6 MRIB view.

   ipv6 multicast routing [**vpn-instance** *vpn-instance-name*]

3. Specify the longest prefix match principle for RPF route selection.

   longest-match

   By default, the route preference principle is used.

Configuring IPv6 multicast load splitting

About this task

You can enable the device to split multiple IPv6 multicast data flows on a per-source basis or on a per-source-and-group basis.

Procedure

1. Enter system view.

   system-view

2. Enter IPv6 MRIB view.

   ipv6 multicast routing [**vpn-instance** *vpn-instance-name*]

3. Configure IPv6 multicast load splitting.

   load-splitting { **source** | **source-group** }

   By default, IPv6 multicast load splitting is disabled.

Configuring an IPv6 multicast forwarding boundary

About this task

You can configure an interface as an IPv6 multicast forwarding boundary for an IPv6 multicast group range. The interface cannot receive or forward IPv6 multicast packets for the groups in the range.

Restrictions and guidelines

You do not need to enable IPv6 multicast routing before this configuration.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure an IPv6 multicast forwarding boundary.

   ipv6 multicast boundary { *ipv6-group-address prefix-length* | **scope** { *scope-id* | **admin-local** | **global** | **organization-local** | **site-local** } }

   By default, an interface is not an IPv6 multicast forwarding boundary for any IPv6 multicast groups.

Setting the maximum number of cached unknown IPv6 multicast packets

About this task

The device caches an IPv6 multicast packet for a period of time if no matching multicast forwarding entry is found for the packet. If a multicast forwarding entry is established for the packet within the time period, the device forwards the packet. This mechanism prevents the device from mistakenly dropping IPv6 multicast packets when the multicast forwarding entries for these packets are to be created.

You can set the maximum number of unknown IPv6 multicast packets that can be cached for an (S, G) entry, in total, or both.

Restrictions and guidelines

As a best practice, set the value in the **ipv6 multicast forwarding-table cache-unknown total** command to be far greater than the value in the **ipv6 multicast forwarding-table cache-unknown per-entry** command.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of unknown IPv6 multicast packets that can be cached for an (S, G) entry.

   ipv6 multicast forwarding-table cache-unknown per-entry *per-entry-limit*

   By default, the device can cache only one unknown IPv6 multicast packet for an (S, G) entry.

3. Set the maximum number of unknown IPv6 multicast packets that can be cached in total.

ipv6 multicast forwarding-table cache-unknown total *total-limit*

By default, the device can cache 1024 unknown IPv6 multicast packets in total.

Configuring an IPv6 MVPN extranet RPF selection policy

About this task

IPv6 MVPN extranet RPF routing policies are used for IPv6 multicast transmission when multicast sources and receivers are located in different VPNs.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB1780, NFNX3-HDB3080	No

Restrictions and guidelines

The PIM modes in the source VPN instance and the receiver VPN instance must be the same. Only PIM-SM and PIM-SSM are supported.

Multicast packets can only be forwarded between two VPNs. The receiver VPN instance cannot also be the source VPN instance.

In PIM-SM mode, you can configure only one RPF selection policy for a multicast group in a VPN instance.

If an IPv6 MVPN extranet RPF selection policy with only the multicast group address specified is configured in the receiver VPN instance, the multicast traffic for the intra-VPN transmission will be interrupted.

To implement source-specific RPF selection in MVPN extranet, you must configure two MVPN extranet RPF routing policies as follows:

- In one policy, specify the address of the RP designated to the multicast group that requires inter-VPN multicast communication as the source address.
- In the other policy, specify the multicast source in the source VPN instance as the source address.

To implement source-and-group-specific RPF selection in MVPN extranet, you must configure two MVPN extranet RPF routing policies as follows:

- In one policy, specify the address of the RP designated to the multicast group as the source address, and specify the multicast group.
- In the other policy, specify the multicast source in the source VPN instance as the source address, and specify the multicast group.
- Make sure the multicast groups in the two policies are the same to avoid inter-VPN multicast transmission failure.

Common Layer 3 multicast supports both the source-PE-based MVPN extranet option and receiver-PE-based MVPN extranet option.

For the source-PE-based MVPN extranet option, if PIM-SM mode is used, the RP of the receiver VPN instance must be configured on the multicast source-side device.

Procedure

1. Enter system view.
   system-view
2. Enter MRIB view.
   multicast routing [**vpn-instance** *vpn-instance-name*]
3. Configure an IPv6 MVPN extranet RPF selection policy.
   ipv6 multicast extranet select-rpf { **vpn-instance** *vpn-instance-name* } { **source** *ipv6-source-address prefix-length* | **group** *ipv6-group-address prefix-length* }*
   By default, no IPv6 MVPN extranet RPF selection policies are configured.

Display and maintenance commands for IPv6 multicast routing and forwarding

△ **CAUTION:**

The **reset** commands might cause IPv6 multicast data transmission failures.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about the interfaces maintained by the IPv6 MRIB.	**display ipv6 mrib** [**vpn-instance** *vpn-instance-name*] **interface** [*interface-type interface-number*]
Display IPv6 multicast boundary information.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **boundary** { **group** [*ipv6-group-address* [*prefix-length*]] | **scope** [*scope-id*] } [**interface** *interface-type interface-number*]
Display IPv6 multicast fast forwarding entries.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **fast-forwarding cache** [*ipv6-source-address* | *ipv6-group-address*] * [**slot** *slot-number*]
Display DF information.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding df-info** [*ipv6-rp-address*] [**verbose**] [**slot** *slot-number*]
Display statistics for IPv6 multicast forwarding events.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding event** [**slot** *slot-number*]
Display IPv6 multicast forwarding entries.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table** [*ipv6-source-address* [*prefix-length*] | *ipv6-group-address* [*prefix-length*] | **incoming-interface** *interface-type interface-number* | **outgoing-interface** { **exclude** | **include** | **match** } *interface-type interface-number* | **slot**

Task	Command
	slot-number | **statistics**] *
Display information about the DF list in the IPv6 multicast forwarding table.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table df-list** [*ipv6-group-address*] [**verbose**] [**slot** *slot-number*]
Display IPv6 multicast routing entries.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **routing-table** [*ipv6-source-address* [*prefix-length*] | *ipv6-group-address* [*prefix-length*] | **incoming-interface** *interface-type interface-number* | **outgoing-interface** { **exclude** | **include** | **match** } *interface-type interface-number*] *
Display RPF information for an IPv6 multicast source.	**display ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **rpf-info** *ipv6-source-address* [*ipv6-group-address*]
Clear IPv6 multicast fast forwarding entries.	**reset ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **fast-forwarding cache** { { *ipv6-source-address* | *ipv6-group-address* } * | **all** } [**slot** *slot-number*]
Clear statistics for IPv6 multicast forwarding events.	**reset ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding event**
Clear IPv6 multicast forwarding entries.	**reset ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **forwarding-table** { { *ipv6-source-address* [*prefix-length*] | *ipv6-group-address* [*prefix-length*] | **incoming-interface** { *interface-type interface-number* } } * | **all** }
Clear IPv6 multicast routing entries.	**reset ipv6 multicast** [**vpn-instance** *vpn-instance-name*] **routing-table** { { *ipv6-source-address* [*prefix-length*] | *ipv6-group-address* [*prefix-length*] | **incoming-interface** *interface-type interface-number* } * | **all** }

NOTE:

- When you clear an IPv6 multicast routing entry, the associated IPv6 multicast forwarding entry is also cleared.
- When you clear an IPv6 multicast forwarding entry, the associated IPv6 multicast routing entry is also cleared.

Contents

Configuring MLD

About MLD

Multicast Listener Discovery (MLD) establishes and maintains IPv6 multicast group memberships between a Layer 3 multicast device and the hosts on the directly connected subnet.

MLD versions

MLD has the following versions:

- MLDv1 (defined by RFC 2710), which is derived from IGMPv2.
- MLDv2 (defined by RFC 3810), which is derived from IGMPv3.

MLDv1 and MLDv2 support the ASM model. MLDv2 can directly implement the SSM model, but MLDv1 must work with the MLD SSM mapping feature to implement the SSM model. For more information about the ASM and SSM models, see "Multicast overview."

How MLDv1 works

MLDv1 implements IPv6 multicast listener management based on the query and response mechanism.

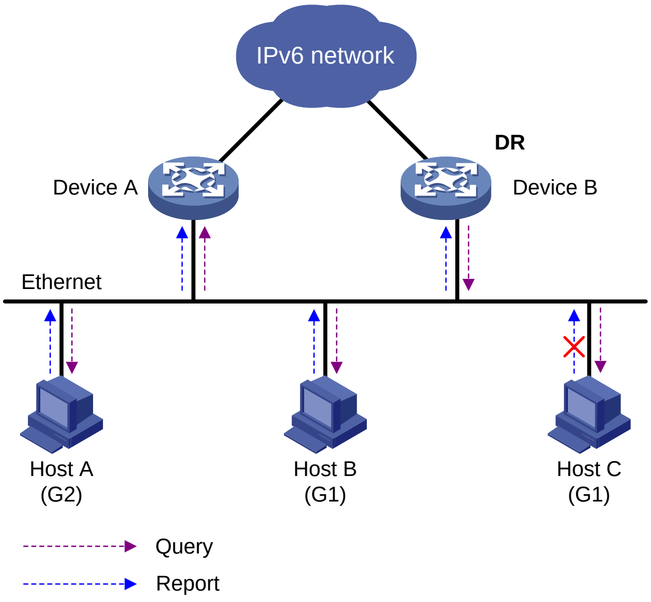
Electing the MLD querier

All IPv6 multicast devices that run MLD on the same subnet can monitor MLD listener report messages (often called reports) from hosts. However, only one device can act as the MLD querier to send MLD query messages (often called queries). A querier election mechanism determines which device acts as the MLD querier on the subnet.

1. Initially, every MLD device assumes itself as the querier. Each device sends MLD general query messages (often called general queries) to all hosts and devices on the local subnet. The destination address of the general queries is FF02::1.
2. After receiving a general query, every MLD device compares the source IPv6 address of the query with its own link-local interface address. The device with the lowest IPv6 address wins the querier election and becomes the querier. All the other devices become non-queriers.
3. All the non-queriers start a timer called the "other querier present timer." If a device receives an MLD query from the querier before the timer expires, it resets this timer. Otherwise, it considers that the querier has timed out. In this case, the device initiates a new querier election process.

Joining an IPv6 multicast group

Figure 1 MLD queries and reports



As shown in Figure 1, Host B and Host C want to receive the IPv6 multicast data addressed to IPv6 multicast group G1. Host A wants to receive the IPv6 multicast data addressed to G2. The following process describes how the hosts join the IPv6 multicast groups and how the MLD querier (Device B in Figure 1) maintains the IPv6 multicast group memberships:

1. The hosts send unsolicited MLD reports to the IPv6 multicast groups they want to join without having to wait for the MLD queries.
2. The MLD querier periodically multicasts MLD queries (with the destination address FF02::1) to all hosts and devices on the local subnet.
3. After receiving a query, the host whose report delay timer expires first sends an MLD report to the IPv6 multicast group G1 to announce its membership for G1. In this example, Host B sends the report. After hearing the report from Host B, Host C, which is on the same subnet as Host B, suppresses its own report for G1.

   Because the MLD devices already know that G1 has a minimum of one member, other members do not need to report their memberships. This mechanism, known as the host MLD report suppression, helps reduce traffic on the local subnet.
4. At the same time, because Host A is interested in G2, it sends a report to the IPv6 multicast group G2.
5. Through the query/report process, the MLD devices determine that G1 and G2 have members on the local subnet. The IPv6 multicast routing protocol (for example, IPv6 PIM) that is running on the devices generates (*, G1) and (*, G2) multicast forwarding entries. These entries are the basis for subsequent IPv6 multicast forwarding. The asterisk (*) represents any IPv6 multicast source.
6. When the IPv6 multicast data addressed to G1 or G2 reaches an MLD device, the device looks up the IPv6 multicast forwarding table. Based on the (*, G1) and (*, G2) entries, the device forwards the IPv6 multicast data to the local subnet. Then, the receivers on the subnet receive the data.

Leaving an IPv6 multicast group

When a host is leaving an IPv6 multicast group, the following process occurs:

1. The host sends an MLD done message to all IPv6 multicast devices on the local subnet. The destination address of done messages is FF02::2.

2. After receiving the MLD done message, the querier sends a configurable number of multicast-address-specific queries to the group that the host is leaving. The IPv6 multicast addresses queried include both the destination address field and the group address field of the message.
3. One of the remaining members (if any on the subnet) in the group sends a report within the time of the maximum response time advertised in the multicast-address-specific queries.
4. If the querier receives a report for the group within the maximum response time, it maintains the memberships of the IPv6 multicast group. Otherwise, the querier assumes that no hosts on the subnet are interested in IPv6 multicast traffic addressed to that group and stops maintaining the memberships of the group.

MLDv2 enhancements

MLDv2 is based on and backwards-compatible with MLDv1. MLDv2 provides hosts with enhanced control capabilities and enhances the MLD state.

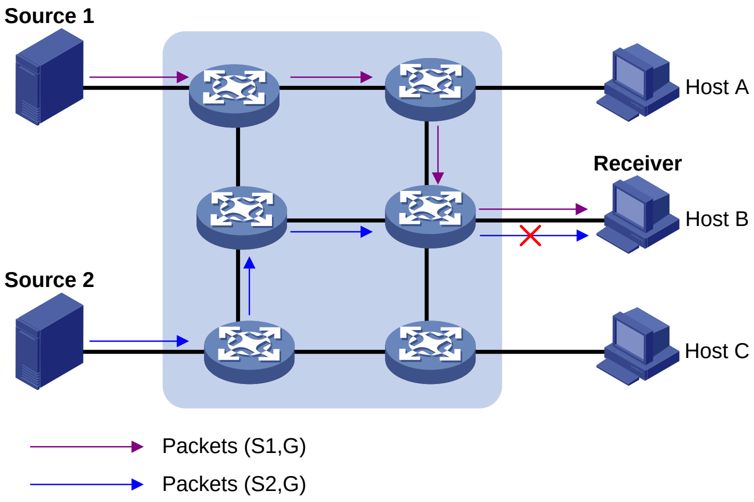
Enhancements in control capability of hosts

MLDv2 has introduced IPv6 multicast source filtering modes (Include and Exclude). These modes allow a host to receive or reject multicast data from the specified IPv6 multicast sources. When a host joins an IPv6 multicast group, one of the following occurs:

- If the host expects IPv6 multicast data from specific IPv6 multicast sources like S1, S2, …, it sends a report with Filter-Mode denoted as "Include Sources (S1, S2, …)."
- If the host does not expect IPv6 multicast data from specific IPv6 multicast sources like S1, S2, …, it sends a report with Filter-Mode denoted as "Exclude Sources (S1, S2, …)."

As shown in Figure 2, the network has two IPv6 multicast sources, Source 1 (S1) and Source 2 (S2). Both of the sources can send IPv6 multicast data to IPv6 multicast group G. Host B wants to receive IPv6 multicast data addressed to G from Source 1 but not from Source 2.

Figure 2 Flow paths of multicast-address-and-source-specific multicast traffic



In MLDv1, Host B cannot select IPv6 multicast sources when it joins IPv6 multicast group G. The IPv6 multicast streams from both Source 1 and Source 2 flow to Host B whether it needs them or not.

In MLDv2, Host B can explicitly express its interest in IPv6 multicast data destined to G from Source 1 but not from Source 2. Then, Host B receives only IPv6 multicast data from Source 1.

Enhancement in MLD state

A multicast device that is running MLDv2 maintains the multicast address state for each multicast address on each attached subnet. The multicast address state consists of the following information:

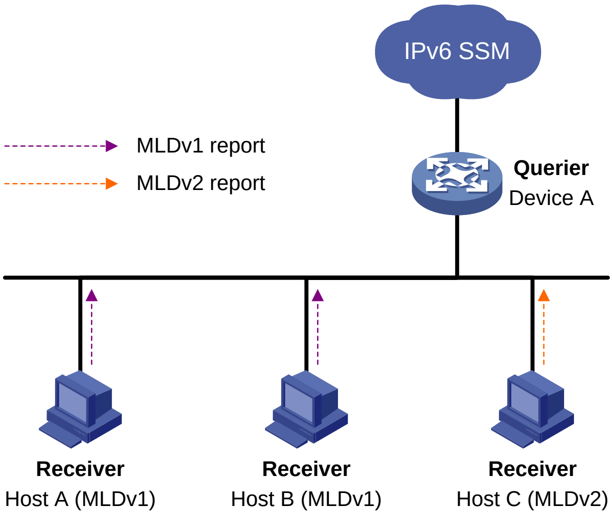
- **Filter mode**—Device keeps tracing the Include or Exclude state.
- **List of sources**—Device keeps tracing the newly added or deleted IPv6 multicast source.
- **Timers**—Filter timers, which include the time that the device waits before switching to the Include mode after an IPv6 multicast address times out, and source timers for source recording.

MLD SSM mapping

An MLDv2 host can explicitly specify multicast sources in its MLDv2 reports. From the reports, the MLD device can obtain the multicast source addresses and directly provide the SSM service. However, an MLDv1 host cannot specify multicast sources in its MLDv1 reports.

The MLD SSM mapping feature enables the MLD device to provide SSM support for MLDv1 receiver host. The device translates (*, G) in MLDv1 reports into (G, INCLUDE, (S1, S2...)) based on the configured MLD SSM mappings.

Figure 3 Network diagram



As shown in Figure 3, Host A and Host B on the IPv6 SSM network run MLDv1, and Host C runs MLDv2. To provide the SSM service for Host A and Host B, you must configure the MLD SSM mapping feature on Device A.

After MLD SSM mappings are configured, Device A checks the IPv6 multicast group address G carried in the message, and performs the following operations:

- If G is not in the IPv6 SSM group range, Device A provides the ASM service.
- If G is in the IPv6 SSM group range but does not match any MLD SSM mapping, Device A drops the report.
- If G is in the IPv6 SSM group range and matches MLD SSM mappings, Device A translates (*, G) in the report to (G, INCLUDE, (S1, S2...)) to provide SSM services.

NOTE:

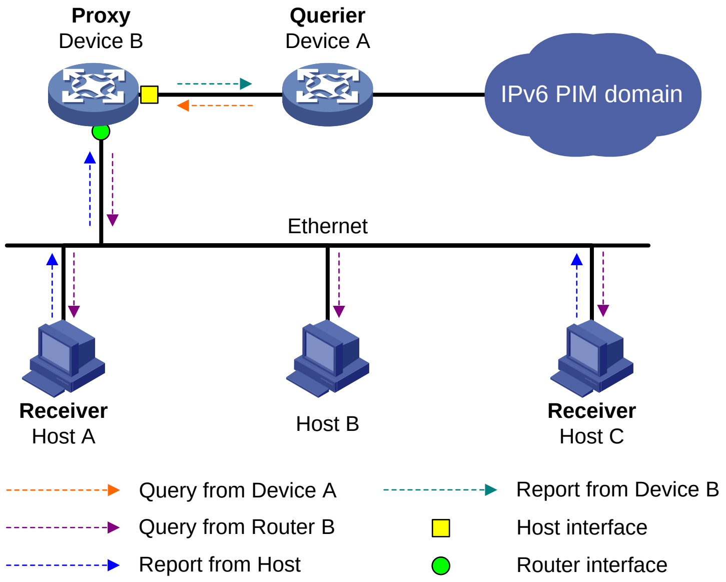
The MLD SSM mapping feature does not process MLDv2 reports.

MLD proxying

As shown in Figure 4, in a simple tree-shaped topology, it is not necessary to configure IPv6 multicast routing protocols, such as IPv6 PIM, on edge devices. Instead, you can configure MLD proxying on these devices. With MLD proxying configured, the edge device acts as an MLD proxy:

- For the upstream MLD querier, the MLD proxy device acts as a host.
- For the downstream receiver hosts, the MLD proxy device acts as an MLD querier.

Figure 4 Network diagram



The following interfaces are defined in MLD proxying:

- **Host interface**—An interface that is in the direction toward the root of the multicast forwarding tree. A host interface acts as a receiver host that is running MLD. MLD proxying must be enabled on this interface. This interface is also called the "proxy interface."
- **Router interface**—An interface that is in the direction toward the leaf of the multicast forwarding tree. A router interface acts as a router that is running MLD. MLD must be configured on this interface.

An MLD proxy device maintains a group membership database, which stores the group memberships on all the router interfaces. The host interfaces and router interfaces perform actions based on this membership database.

- The host interfaces respond to queries according to the membership database or sends join/done messages when the database changes.
- The router interfaces participate in the querier election, send queries, and maintain memberships based on received MLD reports.

MLD support for VPNs

MLD maintains group memberships on a per-interface basis. After receiving an MLD message on an interface, MLD processes the packet within the VPN to which the interface belongs. MLD only communicates with other multicast protocols within the same VPN instance.

Protocols and standards

- RFC 2710, *Multicast Listener Discovery (MLD) for IPv6*
- RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*

MLD tasks at a glance

To configure MLD, perform the following tasks:

1. Enabling MLD
2. (Optional.) Configuring basic MLD features
    - Specifying an MLD version
    - Configuring a static group member
    - Configuring an IPv6 multicast group policy
3. (Optional.) Adjusting MLD performance
    - Configuring MLD query and response parameters
    - Enabling fast-leave processing
4. (Optional.) Configuring MLD SSM mappings
5. (Optional.) Configuring MLD proxying
    - Enabling MLD proxying
    - Enabling IPv6 multicast forwarding on a non-querier interface
    - Enabling IPv6 multicast load splitting for MLD proxy interfaces
6. (Optional.) Enabling MLD NSR

Prerequisites for MLD

Before you configure MLD, you must configure an IPv6 unicast routing protocol so that all devices can interoperate at the network layer.

Enabling MLD

Restrictions and guidelines

Perform this task on interfaces where IPv6 multicast group memberships are created and maintained.

Procedure

1. Enter system view.

    `system-view`

2. Enable IPv6 multicast routing and enter IPv6 MRIB view.

    `ipv6 multicast routing` [**`vpn-instance`** *`vpn-instance-name`*]

    By default, IPv6 multicast routing is disabled.

    For more information about this command, see *IP Multicast Command Reference*.

3. Return to system view.

    `quit`

4. Enter interface view.

    `interface` *`interface-type interface-number`*

5. Enable MLD.

    `mld enable`

    By default, MLD is disabled.

Configuring basic MLD features

Specifying an MLD version

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify an MLD version on the interface.

   `mld version` *`version-number`*

   By default, the MLD version on an interface is MLDv1.

△ **CAUTION:**

For MLD to operate correctly, specify the same MLD version for all devices on the same subnet.

Configuring a static group member

About this task

You can configure an interface as a static member of an IPv6 multicast group. Then, the interface can always receive IPv6 multicast data for the group.

A static group member does not respond to MLD queries. When you complete or cancel this configuration on an interface, the interface does not send an unsolicited MLD report or done message.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Configure a static group member.

   `mld static-group` *`ipv6-group-address`* `[` **`source`** *`ipv6-source-address`* `]`

Configuring an IPv6 multicast group policy

About this task

This feature enables an interface to filter MLD reports by using an ACL that specifies IPv6 multicast groups and the optional sources. It is used to control the IPv6 multicast groups that the hosts attached to an interface can join.

Restrictions and guidelines

This configuration does not take effect on static group members, because static group members do not send MLD reports.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

```
interface interface-type interface-number
```

3. Configure an IPv6 multicast group policy on the interface.

```
mld group-policy ipv6-acl-number [ version-number ]
```

Adjusting MLD performance

Configuring MLD query and response parameters

About this task

The following are MLD query and response parameters:

- **MLD querier's robustness variable**—Number of times for retransmitting MLD queries in case of packet loss. A higher robustness variable makes the MLD querier more robust, but increases the timeout time for IPv6 multicast groups.
- **MLD startup query interval**—Interval at which an MLD querier sends MLD general queries at startup.
- **MLD startup query count**—Number of MLD general queries that an MLD querier sends at startup.
- **MLD general query interval**—Interval at which an MLD querier sends MLD general queries to check for IPv6 multicast group members on the network.
- **MLD last listener query interval**—In MLDv1, it sets the interval at which a querier sends multicast-address-specific queries after receiving a done message. In MLDv2, it sets the interval at which a querier sends multicast-address-and-source-specific queries after receiving a report that changes IPv6 multicast source and group mappings.
- **MLD last listener query count**—In MLDv1, it sets the number of multicast-address-specific queries that the querier sends after receiving a done message. In MLDv2, it sets the number of multicast-address-and-source-specific queries that the querier sends after receiving a report that changes IPv6 multicast group and source mappings.
- **MLD maximum response time**—Maximum time before a receiver responds with a report to an MLD general query. This per-group timer is initialized to a random value in the range of 0 to the maximum response time specified in the MLD query. When the timer value decreases to 0, the receiver sends an MLD report to the group.
- **MLD other querier present timer**—Lifetime for an MLD querier after a non-querier receives an MLD general query. If the non-querier does not receive a new query when this timer expires, the non-querier considers that the querier has failed and starts a new querier election.

Restrictions and guidelines

- To avoid frequent MLD querier changes, set the MLD other querier present timer greater than the MLD general query interval. In addition, configure the same MLD other querier present timer for all MLD devices on the same subnet.
- To speed up the response to MLD queries and avoid simultaneous timer expirations that cause MLD report traffic bursts, you must set an appropriate maximum response time.
  - For MLD general queries, the maximum response time is set by the **`max-response-time`** command.
  - For MLD multicast-address-specific queries or MLD multicast-address-and-source-specific queries, the maximum response time equals the MLD last listener query interval.
- You can configure MLD query and response parameters globally for all interfaces in MLD view or for an interface in interface view. The interface-specific configuration takes priority over the global configuration.

Configuring the MLD query and response parameters globally

1. Enter system view.

   system-view

2. Enter MLD view.

   mld [**vpn-instance** *vpn-instance-name*]

3. Set the MLD querier's robustness variable.

   robust-count *count*

   By default, the MLD querier's robustness variable is 2.

4. Set the MLD startup query interval.

   startup-query-interval *interval*

   By default, the MLD startup query interval is equal to one quarter of the MLD general query interval.

5. Set the MLD startup query count.

   startup-query-count *count*

   By default, the MLD startup query count is equal to the MLD querier's robustness variable.

6. Set the MLD general query interval.

   query-interval *interval*

   By default, the MLD general query interval is 125 seconds.

7. Set the MLD last listener query interval.

   last-listener-query-interval *interval*

   By default, the MLD last listener query interval is 1 second.

8. Set the MLD last listener query count.

   last-listener-query-count *count*

   By default, the MLD last listener query count is equal to the MLD querier's robustness variable.

9. Set the maximum response time for MLD general queries.

   max-response-time *time*

   By default, the maximum response time for MLD general queries is 10 seconds.

10. Set the MLD other querier present timer.

    other-querier-present-timeout *time*

    By default, the MLD other querier present timer is calculated by using the following formula: [MLD general query interval] × [MLD robustness variable] + [maximum response time for MLD general queries] / 2.

Configuring the MLD query and response parameters on an interface

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the MLD querier's robustness variable.

   mld robust-count *count*

   By default, the MLD querier's robustness variable is 2.

4. Set the MLD startup query interval.

   mld startup-query-interval *interval*

   By default, the MLD startup query interval is equal to one quarter of the MLD general query interval.

5. Set the MLD startup query count.

   mld startup-query-count *count*

   By default, the MLD startup query count is equal to the MLD querier's robustness variable.

6. Set the MLD general query interval.

   mld query-interval *interval*

   By default, the MLD general query interval is 125 seconds.

7. Set the MLD last listener query interval.

   mld last-listener-query-interval *interval*

   By default, the MLD last listener query interval is 1 second.

8. Set the MLD last listener query count.

   mld last-listener-query-count *count*

   By default, the MLD last listener query count is equal to the MLD querier's robustness variable.

9. Set the maximum response time for MLD general queries.

   mld max-response-time *time*

   By default, the maximum response time for MLD general queries is 10 seconds.

10. Set the MLD other querier present timer.

    mld other-querier-present-timeout *time*

    By default, the MLD other querier present timer is calculated by using the following formula: [MLD general query interval] × [MLD robustness variable] + [maximum response time for MLD general queries] / 2.

Enabling fast-leave processing

About this task

This feature enables an MLD querier to send leave notifications to the upstream without sending multicast-address-specific or multicast-address-and-source-specific queries after receiving a done message. Use this feature to reduce leave latency and to preserve the network bandwidth.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable fast-leave processing.

   mld fast-leave [**group-policy** *ipv6-acl-number*]

   By default, fast-leave processing is disabled.

Configuring MLD SSM mappings

About this task

This feature enables the device to provide SSM services for MLDv1 hosts.

Restrictions and guidelines

This feature does not process MLDv2 messages. Enable MLDv2 on the receiver-side interface to ensure that MLDv2 reports from MLDv2 receiver hosts can be processed.

Procedure

1. Enter system view.

   system-view

2. Enter MLD view.

   mld [**vpn-instance** *vpn-instance-name*]

3. Configure an MLD SSM mapping.

   ssm-mapping *ipv6-source-address ipv6-acl-number*

Configuring MLD proxying

Prerequisites for MLD proxying

Before you configure MLD proxying, determine the router interfaces and host interface based on the network topology. Then, enable MLD on the router interfaces.

Enabling MLD proxying

Restrictions and guidelines

- Enable MLD proxying on the receiver-side interfaces.
- On an interface enabled with MLD proxying, only the **mld version** command takes effect and other MLD commands do not take effect.
- If you enable both MLD proxying and an IPv6 multicast routing protocol on the same device, the IPv6 multicast routing protocol does not take effect.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable MLD proxying.

   mld proxy enable

   By default, MLD proxying is disabled.

Enabling IPv6 multicast forwarding on a non-querier interface

About this task

Typically, only MLD queriers can forward IPv6 multicast traffic and non-queriers cannot. This prevents IPv6 multicast data from being repeatedly forwarded. If a router interface on the MLD proxy failed the querier election, enable IPv6 multicast forwarding on the interface to forward IPv6 multicast data to downstream receivers.

Restrictions and guidelines

A shared-media network might have multiple MLD proxies, including one proxy acting as a querier. To avoid duplicate IPv6 multicast traffic, do not enable IPv6 multicast forwarding on any of the non-querier MLD proxies for the network.

Procedure

1. Enter system view.

system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Enable IPv6 multicast forwarding on a non-querier interface.

   mld proxy forwarding

   By default, IPv6 multicast forwarding is disabled on a non-querier interface.

Enabling IPv6 multicast load splitting for MLD proxy interfaces

About this task

If multiple MLD proxy interfaces exist on the device, only the proxy interface with the highest IP address forwards IPv6 multicast traffic. You can enable IPv6 multicast load splitting on the device so that all the proxy interfaces can share IPv6 multicast traffic.

Procedure

1. Enter system view.

   system-view

2. Enter MLD view.

   mld [**vpn-instance** *vpn-instance-name*]

3. Enable IPv6 multicast load splitting for MLD proxy interfaces.

   proxy multipath

   By default, IPv6 multicast load splitting is disabled for MLD proxy interfaces.

Enabling MLD NSR

About this task

This feature backs up information about MLD interfaces and MLD multicast groups to the standby process. The device recovers the information without cooperation of other devices when an active/standby switchover occurs. Use this feature to prevent an active/standby switchover from affecting the IPv6 multicast service.

Procedure

1. Enter system view.

   system-view

2. Enable MLD NSR.

   mld non-stop-routing

   By default, MLD NSR is disabled.

Display and maintenance commands for MLD

△ **CAUTION:**

The **reset mld group** command might cause IPv6 multicast data transmission failures.

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about MLD multicast groups.	**display mld** [**vpn-instance** *vpn-instance-name*] **group** [*ipv6-group-address* | **interface** *interface-type interface-number*] [**static** | **verbose**]
Display MLD information for interfaces.	**display mld** [**vpn-instance** *vpn-instance-name*] **interface** [*interface-type interface-number*] [**proxy**] [**verbose**]
Display IPv6 multicast routing entries maintained by the MLD proxy.	**display mld** [**vpn-instance** *vpn-instance-name*] **proxy group** [*ipv6-group-address* | **interface** *interface-type interface-number*] [**verbose**]
Display information about the MLD proxy routing table.	**display mld** [**vpn-instance** *vpn-instance-name*] **proxy routing-table** [*ipv6-source-address* [*prefix-length*] | *ipv6-group-address* [*prefix-length*]] * [**verbose**]
Display MLD SSM mappings.	**display mld** [**vpn-instance** *vpn-instance-name*] **ssm-mapping** *ipv6-group-address*
Clear dynamic MLD multicast group entries.	**reset mld** [**vpn-instance** *vpn-instance-name*] **group** { **all** | **interface** *interface-type interface-number* { **all** | *ipv6-group-address* [*prefix-length*] [*ipv6-source-address* [*prefix-length*]] } }

Troubleshooting MLD

No member information exists on the receiver-side device

Symptom

When a host sends a message to announce that it is joining IPv6 multicast group G, no member information of multicast group G exists on the immediate device.

Solution

To resolve the problem:

1. Use the **display mld interface** command to verify that the networking, interface connections, and IP address configuration are correct.
2. Use the **display current-configuration** command to verify that the IPv6 multicast routing is enabled. If it is not enabled, use the **ipv6 multicast routing** command in system view to enable IPv6 multicast routing. In addition, verify that MLD is enabled on the associated interfaces.

3. Use the **`display mld interface`** command to verify that the MLD version on the interface is lower than that on the host.
4. Use the **`display current-configuration interface`** command to verify that no IPv6 multicast group policies have been configured to filter MLD reports for IPv6 multicast group G.
5. If the problem persists, contact NSFOCUS Support.

Inconsistent membership information on the devices on the same subnet

Symptom

Different memberships are maintained on different MLD devices on the same subnet.

Solution

To resolve the problem:

1. Use the **`display current-configuration`** command to verify the MLD information on the interface. Make sure the devices on the subnet have the same MLD settings on their interfaces.
2. Use the **`display mld interface`** command on all devices on the same subnet to check the MLD timers for inconsistent configurations.
3. Use the **`display mld interface`** command to verify that all devices are running the same MLD version.
4. If the problem persists, contact NSFOCUS Support.

NSFOCUS Firewall Series

NF Network Management and Monitoring Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for network management and monitoring features, including system maintenance and debugging (ping, tracert, and system debugging), NQA, NTP, EAA, process monitoring and maintenance, NETCONF, information center, SNMP, NetStream, RMON, flow log, event MIB, fast log output, mirroring, and CWMP.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.

Convention	Description
#	A line that starts with a pound (#) sign is comments.

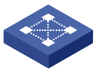
GUI conventions

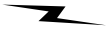
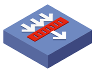
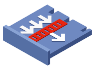
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
❗ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.

Convention	Description
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

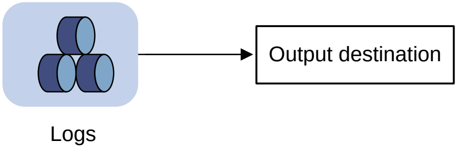
Contents

Configuring the information center

About the information center

The information center on the device receives logs generated by source modules and outputs logs to different destinations according to log output rules. Based on the logs, you can monitor device performance and troubleshoot network problems.

Figure 1 Information center diagram



Log types

Logs are classified into the following types:

- **Standard system logs**—Record common system information. Unless otherwise specified, the term "logs" in this document refers to standard system logs.
- **Diagnostic logs**—Record debugging messages.
- **Security logs**—Record security information, such as authentication and authorization information.
- **Hidden logs**—Record log information not displayed on the terminal, such as input commands.
- **Trace logs**—Record system tracing and debugging messages, which can be viewed only after the devkit package is installed.

Log levels

Logs are classified into eight severity levels from 0 through 7 in descending order. The information center outputs logs with a severity level that is higher than or equal to the specified level. For example, if you specify a severity level of 6 (informational), logs that have a severity level from 0 to 6 are output.

Table 1 Log levels

Severity value	Level	Description
0	Emergency	The system is unusable. For example, the system authorization has expired.
1	Alert	Action must be taken immediately. For example, traffic on an interface exceeds the upper limit.
2	Critical	Critical condition. For example, the device temperature exceeds the upper limit, the power module fails, or the fan tray fails.
3	Error	Error condition. For example, the link state changes.
4	Warning	Warning condition. For example, an interface is disconnected, or the memory resources are used up.
5	Notification	Normal but significant condition. For example, a terminal logs in to the device, or the device reboots.

Severity value	Level	Description
6	Informational	Informational message. For example, a command or a ping operation is executed.
7	Debugging	Debugging message.

Log destinations

The system outputs logs to the following destinations: console, monitor terminal, log buffer, log host, and log file. Log output destinations are independent and you can configure them after enabling the information center. One log can be sent to multiple destinations.

Default output rules for logs

A log output rule specifies the source modules and severity level of logs that can be output to a destination. Logs matching the output rule are output to the destination. Table 2 shows the default log output rules.

Table 2 Default output rules

Destination	Log source modules	Output switch	Severity
Console	All supported modules	Enabled	Debugging
Monitor terminal	All supported modules	Disabled	Debugging
Log host	All supported modules	Enabled	Informational
Log buffer	All supported modules	Enabled	Informational
Log file	All supported modules	Enabled	Informational

Default output rules for diagnostic logs

Diagnostic logs can only be output to the diagnostic log file, and cannot be filtered by source modules and severity levels. Table 3 shows the default output rule for diagnostic logs.

Table 3 Default output rule for diagnostic logs

Destination	Log source modules	Output switch	Severity
Diagnostic log file	All supported modules	Enabled	Debugging

Default output rules for security logs

Security logs can only be output to the security log file, and cannot be filtered by source modules and severity levels. Table 4 shows the default output rule for security logs.

Table 4 Default output rule for security logs

Destination	Log source modules	Output switch	Severity
Security log file	All supported modules	Disabled	Debugging

Default output rules for hidden logs

Hidden logs can be output to the log host, the log buffer, and the log file. Table 5 shows the default output rules for hidden logs.

Table 5 Default output rules for hidden logs

Destination	Log source modules	Output switch	Severity
Log host	All supported modules	Enabled	Informational
Log buffer	All supported modules	Enabled	Informational
Log file	All supported modules	Enabled	Informational

Default output rules for trace logs

Trace logs can only be output to the trace log file, and cannot be filtered by source modules and severity levels. Table 6 shows the default output rules for trace logs.

Table 6 Default output rules for trace logs

Destination	Log source modules	Output switch	Severity
Trace log file	All supported modules	Enabled	Debugging

Log formats and field descriptions

Log formats

The format of logs varies by output destinations. Table 7 shows the original format of log information, which might be different from what you see. The actual format varies by the log resolution tool used.

Table 7 Log formats

Output destination	Format

Console, monitor terminal, log buffer, or log file	`Prefix Timestamp Sysname Module/Level/Mnemonic: Content`
	Example:
	`%Nov 24 14:21:43:502 2016 Sysname SHELL/5/SHELL_LOGIN: VTY logged in from 192.168.1.26`

Log host	**Non-customized format**:
	`<PRI>Timestamp Sysname %%vvModule/Level/Mnemonic: Location; Content`
	Example:
	`<190>Nov 24 16:22:21 2016 Sysname %%10SHELL/5/SHELL_LOGIN: -DevIP=1.1.1.1; VTY logged in from 192.168.1.26<190>Nov 24 16:22:21 2016 Sysname %%10 SHELL/5/SHELL_LOGIN: -DevIP=1.1.1.1; VTY logged in from 192.168.1.26`
	CMCC format:
	`<PRI>Timestamp Sysname %vvModule/Level/Mnemonic: Location; Content`
	Example:
	`<189>Oct 9 14:59:04 2016 Sysname %10SHELL/5/SHELL_LOGIN: VTY logged in from 192.168.1.21`
	SGCC format:
	`<PRI> Timestamp Sysname Devtype Content`

	Example:
	`<189> 2019-03-18 15:39:45 Sysname FW 0 2 admin 10.1.1.1`
	Unicom format:
	`<PRI>Timestamp Hostip vvModule/Level/Serial_number: Content`
	Example:
	`<189>Oct 13 16:48:08 2016 10.1.1.1 10SHELL/5/210231a64jx073000020: VTY logged in from 192.168.1.21`

Log field description

Table 8 Log field description

Field	Description
Prefix (information type)	A log sent to the console, monitor terminal, log buffer, or log file has an identifier in front of the timestamp: • An identifier of percent sign (%) indicates a log with a level equal to or higher than informational. • An identifier of asterisk (*) indicates a debugging log or a trace log. • An identifier of caret (^) indicates a diagnostic log.
PRI (priority)	A log destined for the log host has a priority identifier in front of the timestamp. The priority is calculated by using this formula: facility*8+level, where: • **facility** is the facility name. Facility names local0 through local7 correspond to values 16 through 23. The facility name can be configured using the **`info-center loghost`** command. It is used to identify log sources on the log host, and to query and filter the logs from specific log sources. • **level** is in the range of 0 to 7. See Table 1 for more information about severity levels.
Timestamp	Records the time when the log was generated. Logs sent to the log host and those sent to the other destinations have different timestamp precisions, and their timestamp formats are configured with different commands. For more information, see Table 9 and Table 10.
Hostip	Source IP address of the log. If the **`info-center loghost source`** command is configured, this field displays the IP address of the specified source interface. Otherwise, this field displays the sysname. This field exists only in logs that are sent to the log host in unicom format.
Serial number	Serial number of the master device that generated the log. To view the serial number of the master device, see the DEVICE_SERIAL_NUMBER field in the output of the **`display device manuinfo`** command. This field exists only in logs that are sent to the log host in unicom format.
Sysname (host name or host IP address)	The sysname is the host name or IP address of the device that generated the log. You can use the **`sysname`** command to modify the name of the device.
%% (vendor ID)	Indicates that the information was generated by an NSFOCUS device. This field exists only in logs sent to the log host.
vv (version information)	Identifies the version of the log, and has a value of 10. This field exists only in logs that are sent to the log host.
Module	Specifies the name of the module that generated the log. You can enter the **`info-center source ?`** command in system view to view the module list.
Level	Identifies the level of the log. See Table 1 for more information about severity levels.
Mnemonic	Describes the content of the log. It contains a string of up to 32 characters.

Field	Description
Location	Optional field that identifies the log sender. This field exists only in logs that are sent to the log host in non-customized or CMCC format. The field contains the following information: • **DevIp**—IP address of the log sender. • **Slot**—Member ID of the IRF member device that sent the log. • **SN**—Serial number of the master device that generated the log. To view the serial number of the master device, see the DEVICE_SERIAL_NUMBER field in the output of the **display device manuinfo** command. The **SN** field is available only when the device is configured to add the device serial number to the location field of logs. •
Devtype	Device type. This field exists only in logs that are sent to the log host in SGCC format.
Content	Provides the content of the log. • For most logs, this field is displayed as one or multiple sentences. For example, **VTY logged in from 192.168.1.21.**. • For logs used to record the values of different parameters, this field is displayed in the format of ***key information 1***; ***key information 2***; **…** ***key information n***. Each key information is in the format of ***keyword*(*keyword ID*)=*value*** or ***keyword*(*keyword ID*)=(*description ID*)*description***. A keyword ID identifies a keyword and a description ID identifies a description. The IDs are factory settings to help log host software (such as security management systems) resolve the log key information correctly and efficiently. For example, **streamAlarmType(1032)=(42)Too fast speed of TCP session to destination IP**.

Table 9 Timestamp precisions and configuration commands

Item	Destined for the log host	Destined for the console, monitor terminal, log buffer, and log file
Precision	Seconds	Milliseconds
Command used to set the timestamp format	**info-center timestamp loghost**	**info-center timestamp**

Table 10 Description of the timestamp parameters

Timestamp parameters	Description
boot	Time that has elapsed since system startup, in the format of xxx.yyy. xxx represents the higher 32 bits, and yyy represents the lower 32 bits, of milliseconds elapsed. Logs that are sent to all destinations other than a log host support this parameter. **Example**: `%0.109391473 Sysname FTPD/5/FTPD_LOGIN: User ftp (192.168.1.23) has logged in successfully.` 0.109391473 is a timestamp in the **boot** format.

Timestamp parameters	Description
date	Current date and time. • For logs output to a log host, the timestamp can be in the format of mmm dd hh:mm:ss yyyy (accurate to seconds) or mmm dd hh:mm:ss.ms yyyy (accurate to milliseconds). • For logs output to other destinations, the timestamp is in the format of MMM DD hh:mm:ss:ms YYYY. All logs support this parameter. **Example**: `%May 30 05:36:29:579 2018 Sysname FTPD/5/FTPD_LOGIN: User ftp (192.168.1.23) has logged in successfully.` May 30 05:36:29:579 2018 is a timestamp in the **date** format in logs sent to the console.
iso	Timestamp format stipulated in ISO 8601, accurate to seconds (default) or milliseconds. Only logs that are sent to a log host support this parameter. **Example**: `<189>2018-05-30T06:42:44 Sysname %%10FTPD/5/FTPD_LOGIN: User ftp (192.168.1.23) has logged in successfully.` 2018-05-30T06:42:44 is a timestamp in the **iso** format accurate to seconds. A timestamp accurate to milliseconds is like 2018-05-30T06:42:44.708.
none	No timestamp is included. All logs support this parameter. **Example**: `% Sysname FTPD/5/FTPD_LOGIN: User ftp (192.168.1.23) has logged in successfully.` No timestamp is included.
no-year-date	Current date and time without year information, in the format of MMM DD hh:mm:ss:ms. Only logs that are sent to a log host support this parameter. **Example**: `<189>May 30 06:44:22 Sysname %%10FTPD/5/FTPD_LOGIN: User ftp (192.168.1.23) has logged in successfully.` May 30 06:44:22 is a timestamp in the **no-year-date** format.

Information center tasks at a glance

Managing standard system logs

1. Enabling the information center
2. Outputting logs to various destinations

   Choose the following tasks as needed:
   - Outputting logs to the console
   - Outputting logs to the monitor terminal
   - Outputting logs to log hosts
   - Outputting logs to the log buffer
   - Saving logs to the log file

3. (Optional.) Setting the minimum storage period for logs
4. (Optional.) Enabling synchronous information output
5. (Optional.) Configuring the character set encoding
6. (Optional.) Configuring log suppression

   Choose the following tasks as needed:
   - Enabling duplicate log suppression
   - Configuring log suppression for a module
   - Disabling an interface from generating link up or link down logs
   - Limiting output of security logs
7. (Optional.) Enabling SNMP notifications for system logs

Managing hidden logs

1. Enabling the information center
2. Outputting logs to various destinations

   Choose the following tasks as needed:
   - Outputting logs to log hosts
   - Outputting logs to the log buffer
   - Saving logs to the log file
3. (Optional.) Setting the minimum storage period for logs
4. (Optional.) Enabling duplicate log suppression

Managing security logs

1. Enabling the information center
2. (Optional.) Enabling duplicate log suppression
3. Managing security logs
   - Saving security logs to the security log file
   - Managing the security log file

Managing diagnostic logs

1. Enabling the information center
2. (Optional.) Enabling duplicate log suppression
3. Saving diagnostic logs to the diagnostic log file

Managing trace logs

1. Enabling the information center
2. (Optional.) Enabling duplicate log suppression
3. Setting the maximum size of the trace log file

Enabling the information center

About this task

The information center can output logs only after it is enabled.

Procedure

1. Enter system view.

   system-view

2. Enable the information center.

   info-center enable

   The information center is enabled by default.

Outputting logs to various destinations

Outputting logs to the console

Restrictions and guidelines

The **terminal monitor**, **terminal debugging**, and **terminal logging** commands take effect only for the current connection between the terminal and the device. If a new connection is established, the default is restored.

Procedure

1. Enter system view.

   system-view

2. Configure an output rule for sending logs to the console.

   info-center source { *module-name* | **default** } **console** { **deny** | **level** *severity* }

   For information about default output rules, see "Default output rules for logs."

3. (Optional.) Configure the timestamp format.

   info-center timestamp { **boot** | **date** | **none** }

   The default timestamp format is **date**.

4. Return to user view.

   quit

5. Enable log output to the console.

   terminal monitor

   By default, log output to the console is enabled.

6. Enable output of debugging messages to the console.

   terminal debugging

   By default, output of debugging messages to the console is disabled.

   This command enables output of debugging-level log messages to the console.

7. Set the lowest severity level of logs that can be output to the console.

   terminal logging level *severity*

   The default setting is 6 (informational).

Outputting logs to the monitor terminal

About this task

Monitor terminals refer to terminals that log in to the device through the VTY line.

Restrictions and guidelines

The **terminal monitor**, **terminal debugging**, and **terminal logging** commands take effect only for the current connection between the terminal and the device. If a new connection is established, the default is restored.

Procedure

1. Enter system view.

   system-view

2. Configure an output rule for sending logs to the monitor terminal.

   info-center source { *module-name* | **default** } **monitor** { **deny** | **level** *severity* }

   For information about default output rules, see "Default output rules for logs."

3. (Optional.) Configure the timestamp format.

   info-center timestamp { **boot** | **date** | **none** }

   The default timestamp format is **date**.

4. Return to user view.

   quit

5. Enable log output to the monitor terminal.

   terminal monitor

   By default, log output to the monitor terminal is disabled.

6. Enable output of debugging messages to the monitor terminal.

   terminal debugging

   By default, output of debugging messages to the monitor terminal is disabled.

   This command enables output of debugging-level log messages to the monitor terminal.

7. Set the lowest level of logs that can be output to the monitor terminal.

   terminal logging level *severity*

   The default setting is 6 (informational).

Outputting logs to log hosts

Restrictions and guidelines

The device supports the following methods (in descending order of priority) for outputting logs of a module to designated log hosts:

- Fast log output.

  For information about the modules that support fast log output and how to configure fast log output, see "Configuring fast log output."

- Flow log.

  For information about the modules that support flow log output and how to configure flow log output, see "Configuring flow log."

- Information center.

If you configure multiple log output methods for a module, only the method with the highest priority takes effect.

Procedure

1. Enter system view.

   system-view

2. Configure a log output filter or a log output rule. Choose one option as needed:

   - Configure a log output filter.

     info-center filter *filter-name* { *module-name* | **default** } { **deny** | **level** *severity* }

     By default, no log output filters exist.

     You can create multiple log output filters. When specifying a log host, you can apply a log output filter to the log host to control log output.

   - Configure a log output rule for the log host output destination.

     info-center source { *module-name* | **default** } **loghost** { **deny** | **level** *severity* }

     For information about default output rules, see "Default output rules for logs."

   The system chooses the settings to control log output to a log host in the following order:

   a. Log output filter applied to the log host.

   b. Log output rules configured for the log host output destination by using the **info-center source** command.

   c. Default log output rules (see "Default output rules for logs").

3. (Optional.) Specify a source IP address for output logs.

   info-center loghost source *interface-type interface-number*

   By default, the source IP address of output logs is the primary IP address of their outgoing interfaces.

4. (Optional.) Specify the format in which logs are output to log hosts.

info-center format { **cmcc** | **sgcc** | **unicom** }

   By default, logs are output to log hosts in non-customized format.

5. (Optional.) Add the device serial number to the location field of logs sent to log hosts.

   info-center loghost locate-info with-sn

   By default, the device does not add the device serial number to the location field of logs sent to log hosts.

6. (Optional.) Configure the timestamp format.

   info-center timestamp loghost { **date** [**with-milliseconds**] | **iso** [**with-milliseconds**] | **no-year-date** | **none** }

   The default timestamp format is **date**.

7. Specify a log host and configure related parameters.

   info-center loghost [**vpn-instance** *vpn-instance-name*] { *hostname* | *ipv4-address* | **ipv6** *ipv6-address* } [**facility** *local-number* | **filter** *filter-name* | **format** { **cmcc** | **default** | **sgcc** | **unicom** } | **port** *port-number* | **source-ip** *source-ip-address*] *

   By default, no log hosts or related parameters are specified.

   The value for the *port-number* argument must be the same as the value configured on the log host. Otherwise, the log host cannot receive logs.

The log format specified by this command has a higher priority than that specified by the **info-center format** command. The source IP address specified by this command has a higher priority than that specified by the **info-center loghost source** command.

Outputting logs to the log buffer

About this task

This feature enables log output to log buffers based on the log source modules. Log buffers include the following types:

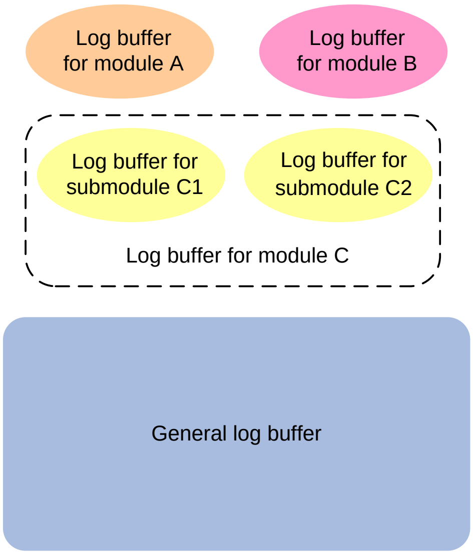
- **Separate module-specific log buffers**—Logs generated by modules or submodules that have separate log buffers are saved to their respective log buffers.

  For example, session logs and attack defense logs are saved to the session log buffer and the attack defense log buffer, respectively. With separate buffers, you can manage and view the module-specific logs and general logs conveniently.

  The modules and submodules that have separate log buffers are determined by factory settings. To view the names of supported modules and submodules, use the **display logbuffer module ?** command and the **display logbuffer module** *module-name* **submodule ?** command, respectively. You can use the **display logbuffer module** *module-name* to view buffered logs of the specified module. You can use the **display logbuffer module** *module-name* and **submodule** *submodule-name* command to view buffered logs of the specified submodule.
- **General log buffer**—Logs generated by the system and the modules other than the modules and submodules that have separate log buffers are saved to the general log buffer.

  You can use the **display logbuffer** command without the **module** keyword to view buffered logs in the general log buffer.

Figure 2 Genratal log buffer and separate log buffers for modules and submodules



Procedure

1. Enter system view.

   system-view
2. (Optional.) Configure an output rule for sending logs to the log buffer.

   info-center source { *module-name* | **default** } **logbuffer** { **deny** | **level** *severity* }

For information about the default output rules, see "Default output rules for logs."

3. (Optional.) Configure the timestamp format.

   info-center timestamp { **boot** | **date** | **none** }

   The default timestamp format is **date**.

4. Enable log output to the log buffer.

   info-center logbuffer

   By default, log output to the log buffer is enabled.

5. (Optional.) Set the maximum size of the log buffer for a module.

   info-center logbuffer module *module-name* **size** *buffersize*

   By default, a maximum of 512 logs can be buffered.

6. (Optional.) Set the maximum size of the general log buffer.

   info-center logbuffer size *buffersize*

   By default, a maximum of 512 logs can be buffered.

Saving logs to the log file

About this task

The log file feature enables log output to log files based on the log source modules. Log files include the following types:

- **Separate module-specific log files**—Logs generated by modules or submodules that have separate log files are output to their respective log files.

  For example, session logs and attack defense logs are output to the session log file and the attack defense log file, respectively. With separate log files, you can manage and view the module-specific logs and general logs conveniently.

  The modules that have separate log files are determined by factory settings. To view the names of supported modules, use the **info-center logfile module ?** command. You can use the **more** command in user view to logs in a module-specified log file.

- **General log file**—Logs generated by the system and the modules other than the modules that have separate log files are output to the general log file.

The system saves logs from the log file buffer to the log file. After saving logs to the log file, the system clears the log file buffer.

Log saving from the log file buffer to the log file is triggered by the following situations:

- Periodic saving.
- A manual immediate saving of buffered logs to the log file.
- The log file buffer is full.

A log file is automatically created when needed and has a maximum capacity. When log file overwrite-protection is disabled and no log file space or storage device space is available, the system will replace the oldest logs with new logs.

TIP:

- Clean up the storage space of the device regularly to ensure sufficient storage space for the log file feature.
- As a best practice, back up the log files regularly to avoid loss of important logs.

Configuring a module-specific log file

1. Enter system view.

system-view

2. Configure an output rule for sending logs to the log file.

   info-center source { *module-name* | **default** } **logfile** { **deny** | **level** *severity* }

   For information about default output rules, see "Default output rules for logs."

3. Enable the log file feature.

   info-center logfile enable

   By default, the log file feature is enabled.

   This command takes effect on both module-specific log files and the general log file.

4. (Optional.) Set the maximum log file size.

   info-center logfile module *module-name* **size-quota** *size*

   By default, the maximum log file size for the general log file and module-specific log file is 10 MB and 1 MB, respectively.

5. (Optional.) Set the alarm threshold for log file usage of a specific module.

   info-center logfile module *module-name* **alarm-threshold** *usage*

   The default alarm threshold for log file usage ratio is 80%. When the usage ratio of the log file reaches 80%, the system outputs a message to inform the user.

   Setting the alarm threshold to 0 means to disable the log file usage alarm feature for the module.

6. (Optional.) Specify the log file directory.

   info-center logfile directory *dir-name*

   By default, the log file is saved in the **logfile** folder under the root directory of the default file system.

   This command takes effect on both independent module log files and the general log file.

   This command cannot survive an IRF reboot or a master/subordinate switchover.

7. Save logs in the log file buffer to the log file. Choose one or both options as needed:

   - Configure the automatic log file saving interval.

     info-center logfile frequency *freq-sec*

     The default log file saving interval is 86400 seconds.

     This command takes effect on both module-specific log files and the general log file.

   - Immediately save logs from the log file buffer to the log file.

     logfile save

     This command is available in any view.

     This command takes effect on both module-specific log files and the general log file.

Configuring the general log file

1. Enter system view.

   system-view

2. (Optional.) Configure an output rule for sending logs to the log file.

   info-center source { *module-name* | **default** } **logfile** { **deny** | **level** *severity* }

   For information about the default output rules, see "Default output rules for logs."

3. Enable the log file feature.

   info-center logfile enable

   By default, the log file feature is enabled.

   This command takes effect on both module-specific log files and the general log file.

4. (Optional.) Set the maximum log file size.

   ```
   info-center logfile size-quota size
   ```

   By default, the maximum log file size for the general log file and module-specific log file is 10 MB and 1 MB, respectively.

5. (Optional.) Specify the log file directory.

   ```
   info-center logfile directory dir-name
   ```

   By default, the log file is saved in the **logfile** folder under the root directory of the default file system.

   This command takes effect on both module-specific log files and the general log file.

   This command cannot survive an IRF reboot or a master/subordinate switchover.

6. Save logs in the log file buffer to the log file. Choose one or both options as needed:

   - Configure the automatic log file saving interval.

     ```
     info-center logfile frequency freq-sec
     ```

     The default log file saving interval is 86400 seconds.

     This command takes effect on both independent module log files and the general log file.

   - Immediately save logs from the log file buffer to the log file.

     ```
     logfile save
     ```

     This command is available in any view.

     This command takes effect on both module-specific log files and the general log file.

Setting the minimum storage period for logs

About setting the minimum storage period

Use this feature to set the minimum storage period for logs in log buffers and log files. This feature ensures that logs will not be overwritten by new logs during a set period of time.

Logs of the modules that have separate log buffers and log files are output to their respective log buffers or log files. Logs of other modules and the system logs are output the general log buffer or log file. By default, when a log buffer or log file is full, new logs will automatically overwrite the oldest logs. After the minimum storage period is set, the system identifies the storage period of a log to determine whether to delete the log. The system current time minus a log's generation time is the log's storage period.

- If the storage period of a log is shorter than or equal to the minimum storage period, the system does not delete the log. The new log will not be saved.
- If the storage period of a log is longer than the minimum storage period, the system deletes the log to save the new log.

Setting the minimum storage period for logs in a module-specific log buffer or log file

1. Enter system view.

   ```
   system-view
   ```

2. Set the minimum storage period for logs stored in the log buffer or log file of a module.

   ```
   info-center syslog module module-name min-age min-age
   ```

   By default, the log minimum storage period is not set.

Setting the minimum storage period for logs in the general log buffer and log file

1. Enter system view.

   system-view

2. Set the minimum storage period for logs stored in the general log buffer and log file.

   info-center syslog min-age *min-age*

   By default, the log minimum storage period is not set.

Enabling synchronous information output

About this task

System log output interrupts ongoing configuration operations, obscuring previously entered commands. Synchronous information output shows the obscured commands. It also provides a command prompt in command editing mode, or a [Y/N] string in interaction mode so you can continue your operation from where you were stopped.

Procedure

1. Enter system view.

   system-view

2. Enable synchronous information output.

   info-center synchronous

   By default, synchronous information output is disabled.

Configuring the character set encoding used on the information center

About this task

The information center supports outputting log messages by using the GB18030 or UTF-8 encoding. By default, the GB18030 encoding is used.

For the login terminal to correctly display Chinese characters in log messages received from the information center, the information center and the terminal must use the same character set encoding.

Procedure

1. Enter system view.

   system-view

2. Enable the information center to use the UTF-8 encoding.

   info-center syslog utf-8 enable

   By default, the information center uses the GB18030 encoding for outputting log messages.

Configuring log suppression

Enabling duplicate log suppression

About this task

Output of consecutive duplicate logs (logs that have the same module name, level, mnemonic, location, and text) wastes system and network resources.

With duplicate log suppression enabled, the system starts a suppression period upon outputting a log:

- If only duplicate logs are received during the suppression period, the information center does not output the duplicate logs. When the suppression period expires, the information center outputs the suppressed log and the number of times the log is suppressed.
- If a different log is received during the suppression period, the information center performs the following operations:
  - Outputs the suppressed log and the number of times the log is suppressed.
  - Outputs the different log and starts a suppression period for that log.
- If no log is received within the suppression period, the information center does not output any message when the suppression period expires.

Procedure

1. Enter system view.

   `system-view`

2. Enable duplicate log suppression.

   `info-center logging suppress duplicates`

   By default, duplicate log suppression is disabled.

Configuring log suppression for a module

About this task

This feature suppresses output of logs. You can use this feature to filter out the logs that you are not concerned with.

Perform this task to configure a log suppression rule to suppress output of all logs or logs with a specific mnemonic value for a module.

Procedure

1. Enter system view.

   `system-view`

2. Configure a log suppression rule for a module.

   `info-center logging suppress module` *`module-name`* **`mnemonic`** { **`all`** | *`mnemonic-value`* }

   By default, the device does not suppress output of any logs from any modules.

Disabling an interface from generating link up or link down logs

About this task

By default, an interface generates link up or link down log information when the interface state changes. In some cases, you might want to disable certain interfaces from generating this information. For example:

- You are concerned about the states of only some interfaces. In this case, you can use this function to disable other interfaces from generating link up and link down log information.
- An interface is unstable and continuously outputs log information. In this case, you can disable the interface from generating link up and link down log information.

Use the default setting in normal cases to avoid affecting interface status monitoring.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Disable the interface from generating link up or link down logs.

   ```
   undo enable log updown
   ```

   By default, an interface generates link up and link down logs when the interface state changes.

Limiting output of security logs

About this task

When the device processes the following security services, a large number of security logs will be generated and sent to the information center, which will cause overload of the information center:

- AFT.
- ASPF.
- Data filtering.
- File filtering.
- URL filtering.
- NAT.
- Session management.
- Anti-virus.
- Application audit and management.
- IPS.
- NetShare Control.
- Server connection detection.
- Attack detection and prevention.

This feature allows you to set the maximum number of security logs that can be sent per second to the information center. With this feature configured, the device will discard the subsequent security logs when the maximum number of security logs that can be sent per second is reached. Please configure the maximum number of security logs that can be sent per second as required.

Procedure

1. Enter system view.

   system-view

2. Set the maximum number of security logs that can be sent per second.

   security syslog rate-limit *max-value*

   By default, the device can send a maximum of 1000 security logs per second to the information center.

Enabling SNMP notifications for system logs

About this task

This feature enables the device to send an SNMP notification for each log message it outputs. The device encapsulates the logs in SNMP notifications and then sends them to the SNMP module and the log trap buffer.

You can configure the SNMP module to send received SNMP notifications in SNMP traps or informs to remote hosts. For more information, see "Configuring SNMP."

To view the traps in the log trap buffer, access the MIB corresponding to the log trap buffer.

Procedure

1. Enter system view.

   system-view

2. Enable SNMP notifications for system logs.

   snmp-agent trap enable syslog

   By default, the device does not send SNMP notifications for system logs.

3. Set the maximum number of traps that can be stored in the log trap buffer.

   info-center syslog trap buffersize *buffersize*

   By default, the log trap buffer can store a maximum of 1024 traps.

Managing security logs

Saving security logs to the security log file

About this task

Security logs are very important for locating and troubleshooting network problems. Generally, security logs are output together with other logs. It is difficult to identify security logs among all logs.

To solve this problem, you can save security logs to the security log file without affecting the current log output rules.

After you enable the security log file feature, the system processes security logs as follows:

1. Outputs security logs to the security log file buffer.
2. Saves logs from the security log file buffer to the security log file at the specified interval.

   If you have the security-audit role, you can also manually save security logs to the security log file.

3. Clears the security log file buffer immediately after the security logs are saved to the security log file.

Restrictions and guidelines

The device supports only one security log file. The system will overwrite old logs with new logs when the security log file is full. To avoid security log loss, you can set an alarm threshold for the security log file usage ratio. When the alarm threshold is reached, the system outputs a message to inform you of the alarm. You can log in to the device with the security-audit user role and back up the security log file to prevent the loss of important data.

Procedure

1. Enter system view.

   `system-view`

2. Enable the security log file feature.

   `info-center security-logfile enable`

   By default, the security log file feature is disabled.

3. Set the interval at which the system saves security logs.

   `info-center security-logfile frequency` *`freq-sec`*

   The default security log file saving interval is 86400 seconds.

4. (Optional.) Set the maximum size for the security log file.

   `info-center security-logfile size-quota` *`size`*

   By default, the maximum security log file size is 10 MB.

5. (Optional.) Set the alarm threshold of the security log file usage.

   `info-center security-logfile alarm-threshold` *`usage`*

   By default, the alarm threshold of the security log file usage ratio is 80. When the usage of the security log file reaches 80%, the system will send a message.

Managing the security log file

Restrictions and guidelines

To use the security log file management commands, you must have the security-audit user role. For information about configuring the security-audit user role, see AAA in *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Change the directory of the security log file.

   `info-center security-logfile directory` *`dir-name`*

   By default, the security log file is saved in the **seclog** directory in the root directory of the storage device.

   This command cannot survive an IRF reboot or a master/subordinate switchover.

3. Manually save all logs in the security log file buffer to the security log file.

   `security-logfile save`

   This command is available in any view.

4. (Optional.) Display the summary of the security log file.

   `display security-logfile summary`

   This command is available in any view.

Saving diagnostic logs to the diagnostic log file

About this task

By default, the diagnostic log file feature saves diagnostic logs from the diagnostic log file buffer to the diagnostic log file at the specified saving interval. You can also manually trigger an immediate saving of diagnostic logs to the diagnostic log file. After saving diagnostic logs to the diagnostic log file, the system clears the diagnostic log file buffer.

The device supports only one diagnostic log file. The diagnostic log file has a maximum capacity. When the capacity is reached, the system replaces the oldest diagnostic logs with new logs.

Procedure

1. Enter system view.

   system-view

2. Enable the diagnostic log file feature.

   info-center diagnostic-logfile enable

   By default, the diagnostic log file feature is enabled.

3. (Optional.) Set the maximum diagnostic log file size.

   info-center diagnostic-logfile quota *size*

   By default, the maximum diagnostic log file size is 10 MB.

4. (Optional.) Specify the diagnostic log file directory.

   info-center diagnostic-logfile directory *dir-name*

   By default, the diagnostic log file is saved in the **diagfile** folder under the root directory of the default file system.

   This command cannot survive an IRF reboot or a master/subordinate switchover.

5. Save diagnostic logs in the diagnostic log file buffer to the diagnostic log file. Choose one option as needed:

   - Configure the automatic diagnostic log file saving interval.

     info-center diagnostic-logfile frequency *freq-sec*

     The default diagnostic log file saving interval is 86400 seconds.

   - Manually save diagnostic logs to the diagnostic log file.

     diagnostic-logfile save

     This command is available in any view.

Setting the maximum size of the trace log file

About this task

The device has only one trace log file. When the trace log file is full, the device overwrites the oldest trace logs with new ones.

Procedure

1. Enter system view.

   system-view

2. Set the maximum size for the trace log file.

   info-center trace-logfile quota *size*

   The default maximum trace log file size is 1 MB.

Display and maintenance commands for information center

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display the character set encoding used on the device or the login terminal.	**display character-set** [**terminal**]
Display the diagnostic log file configuration.	**display diagnostic-logfile summary**
Display the information center configuration.	**display info-center**
Display information about log output filters.	**display info-center filter** [*filtername*]
Display the log output rules by source modules.	**display info-center source** [**module** *module-name*]
Display log buffer information and buffered logs.	**display logbuffer** [**module** *module-name* [**submodule** *submodule-name*]] [**reverse**] [**level** *severity* | **size** *buffersize* | **slot** *slot-number*] * [**last-mins** *mins*]
Display the log buffer summary.	**display logbuffer summary** [**level** *severity* | **slot** *slot-number*] *
Display the log file configuration.	**display logfile summary**
Display summary information of the security log file (security-audit user role required).	**display security-logfile summary**
Clear the log buffer.	**reset logbuffer** [**module** *module-name* [**submodule** *submodule-name*]]

Contents

Configuring flow log

About flow log

Flow log records session information based on flows.

A flow log entry might contain the following information about a flow:

- Network access quintuple information (source IP address, destination IP address, source port, destination port, and protocol number).
- Statistics about sent and received packets.
- Flow-based link selection information.

Flow log export

You can export flow log entries in the following methods:

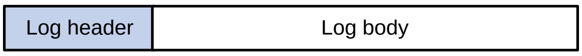
- Export flow log entries to log hosts. Flow log entries are sent as binary characters in UDP. One UDP packet can contain multiple log entries.
- Export flow log entries to the information center. Flow log entries are converted to syslog entries in ASCII format, with the informational severity level. The information center specifies the output destinations for the log entries. For more information about the information center, see "Configuring the information center."

Log entries in ASCII format are human readable. However, the log data volume is higher in ASCII format than in binary format. It's recommended to export flow log entries in binary format to log hosts if the log data volume is large.

Flow log packet

The flow log data (including the log header and log body) is carried in the body of a UDP packet. The log header contains the common fields and the log body contains the service-specific log information. The log body of a flow log packet can contain one or multiple log entries of the same type.

Figure 1 Flow log packet



Flow log supports multiple service modules. The log body fields vary with the log type or log version. For more information about flow log fields, see "Appendix A Flow log fields."

Flow log tasks at a glance

To configure flow log, perform the following tasks:

1. Enabling flow log
2. Specifying a flow log export destination

   Choose one of the following tasks:

   - Specifying a log host as the flow log export destination
   - Specifying the information center as the flow log export destination

3. (Optional.) Configuring the flow log version
4. (Optional.) Specifying a source IP address for flow log packets
5. (Optional.) Configuring the timestamp of flow log entries
6. (Optional.) Enabling load balancing for flow log entries
7. (Optional.) Configuring flow log host groups

Enabling flow log

Before you configure the flow log feature, complete the following tasks to enable flow log output for modules:

- Enable NAT flow log output by using the **nat log enable** command. For more information about the NAT logging commands, see *NAT Command Reference*.
- Enable AFT flow log output by using the **aft log enable** command. For more information, see AFT commands in *Layer 3—IP Services Command Reference*.
- Enable load balancing NAT flow log output by using the **loadbalance log enable nat** command. For more information, see load balancing commands in *Load Balancing Command Reference*.

Specifying a flow log export destination

Restrictions and guidelines for flow log export destination configuration

You can export flow log entries to a log host or to the information center, but not both. If you configure both methods, the system exports flow log entries to the information center.

flow log entries exported to the information center has the **informational** severity level.

Specifying a log host as the flow log export destination

1. Enter system view.

   system-view

2. Specify a log host as the destination for flow log export.

   userlog flow export [**vpn-instance** *vpn-instance-name*] **host** { *hostname* | *ipv4-address* | **ipv6** *ipv6-address* } **port** *udp-port*

   By default, no log hosts are specified.

   You can specify multiple log hosts.

Specifying the information center as the flow log export destination

1. Enter system view.

   system-view

2. Specify the information center as the destination for flow log export.

   userlog flow syslog

   By default, flow log entries are not exported to the information center.

△ **CAUTION:**

The device might generate a lot of session logs in a short time. If the session logs are exported to the information center for processing, the information center might consume too much performance, affecting normal operations of the device.

Configuring the flow log version

Restrictions and guidelines

Make sure the specified flow log version is supported on the log host.

If you configure the flow log version multiple times, the most recent configuration takes effect.

Procedure

1. Enter system view.

   `system-view`

2. Configure the flow log version.

   `userlog flow export version` *`version-number`*

   The default flow log version is 1.0.

Specifying a source IP address for flow log packets

About this task

By default, the source IP address for flow log packets is the IP address of their outgoing interface. For the log hosts to filter log entries by log sender, specify a source IP address for all flow log packets.

Restrictions and guidelines

As a best practice, use a Loopback interface's address as the source IP address for flow log packets. A Loopback interface is always up. The setting avoids export failure on interfaces that might go down.

Procedure

1. Enter system view.

   `system-view`

2. Specify a source IP address for flow log packets.

   `userlog flow export source-ip` { *`ipv4-address`* | **`ipv6`** *`ipv6-address`* }

   By default, the source IP address for flow log packets is the IP address of their outgoing interface.

Configuring the timestamp of flow log entries

About this task

The device uses either the local time or the UTC time in the timestamp of flow logs.

- **UTC time**—Standard Greenwich Mean Time (GMT).
- **Local time**—Standard GMT plus or minus the time zone offset.

The time zone offset can be configured by using the **`clock timezone`** command. For more information, see device management in *Fundamentals Command Reference*.

Procedure

1. Enter system view.

```
system-view
```

2. Configure the device to use the local time in the flow log timestamp.

```
userlog flow export timestamp localtime
```

By default, the UTC time is used in the flow log timestamp.

Enabling load balancing for flow log entries

About this task

By default, the device sends a copy of each flow log entry to all configured log hosts. When one log host fails, other log hosts still have complete flow log entries.

In load balancing mode, flow log entries are distributed among log hosts based on the source IP addresses (before NAT) that are recorded in the entries. The flow log entries generated for the same source IP address are sent to the same log host.

Restrictions and guidelines

In load balancing mode, flow log entries are load balanced among all configured log hosts, regardless of whether the log hosts are reachable. If a log host is unreachable, the flow log entries sent to it will be lost.

Procedure

1. Enter system view.

```
system-view
```

2. Enable load balancing for flow log entries.

```
userlog flow export load-balancing
```

By default, load balancing is disabled.

Configuring flow log host groups

About flow log host group

By default, the device sends a copy of each flow log entry to all available log hosts. To filter logs and reduce the log sending and processing workload of the device, configure the flow log host group feature.

The flow log host group feature allows you to classify flow log hosts into groups and specify an ACL for each group. A flow log matches a log host group if it matches the group's ACL, and it is sent only to the log hosts in the matching group.

If a flow log matches multiple log host groups, the device sends the log to the group that comes first in alphabetical order of the matching group names.

If a flow log does not match any log host groups, the device ignores the log host group configuration and sends the log to all configured log hosts.

If load balancing is enabled, flow logs sent to a log host group will be load-shared among the log hosts in the group. flow logs generated for the same source IP address are sent to the same log host.

Prerequisites for log host group configuration

Before you configure flow log host groups, complete the following tasks:

- Configure the ACLs to be used by the flow log host groups.
- Use the **userlog flow export host** command to configure the log hosts to be assigned to the flow log host groups.

Configuring an IPv4 flow log host group

1. Enter system view.

   system-view
2. Create an IPv4 flow log host group and enter its view.

   userlog host-group *host-group-name* **acl** { **name** *acl-name* | **number** *acl-number* }

   By default, no IPv4 flow log host groups exist.
3. Assign an IPv4 log host to the flow log host group.

   userlog host-group [**vpn-instance** *vpn-instance-name*] **host flow** { *hostname* | *ipv4-address* }

   By default, an IPv4 flow log host group does not contain any log hosts.

Configuring an IPv6 flow log host group

1. Enter system view.

   system-view
2. Create an IPv6 flow log host group and enter its view.

   userlog host-group ipv6 *host-group-name* **acl** { **name** *acl-name* | **number** *acl-number* }

   By default, no IPv6 flow log host groups exist.
3. Assign an IPv6 log host to the flow log host group.

   userlog host-group [**vpn-instance** *vpn-instance-name*] **host flow ipv6** { *hostname* | *ipv6-address* }

   By default, an IPv6 flow log host group does not contain any log hosts.

Display and maintenance commands for flow log

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display flow log configuration and statistics.	**display userlog export**
Display flow log host group information.	**display userlog host-group** [**ipv6**] [*host-group-name*]
Clear flow log statistics.	**reset userlog flow export**

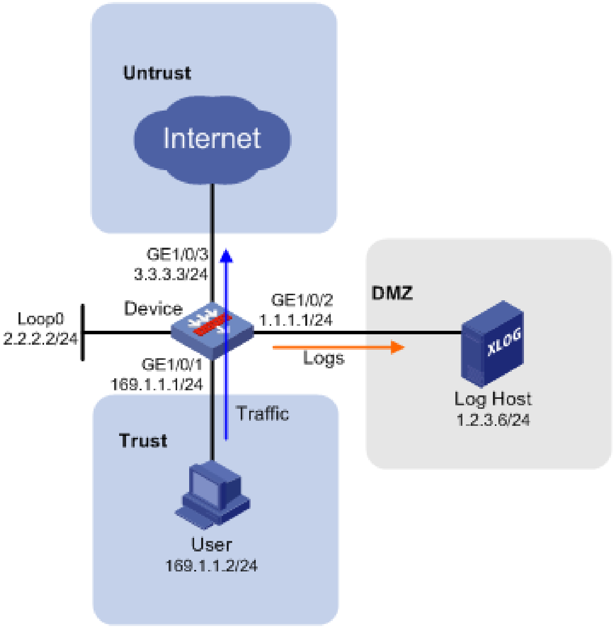
NAT flow log configuration examples

Example: Configuring NAT flow log export

Network configuration

As shown in Figure 2, configure flow log on the device to send flow log entries generated for the user to the log host.

Figure 2 Network diagram



Procedure

1. Assign IP addresses to interfaces.

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface loopback 0
   [Device-LoopBack0] ip address 2.2.2.2 255.255.255.0
   [Device-LoopBack0] quit
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 169.1.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hops in the routes are 1.1.1.2 and 3.3.3.1, respectively.

   ```
   [Device] ip route-static 1.2.3.0 24 1.1.1.2
   [Device] ip route-static 0.0.0.0 0 3.3.3.1
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   ```

```
[Device-security-zone-Trust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-security-zone-Untrust] quit
```

4. Configure a security policy:

   # Configure a rule named **loglocalout** to allow the device to send log packets to the log host.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name loglocalout
   [Device-security-policy-ip-1-loglocalout] source-zone local
   [Device-security-policy-ip-1-loglocalout] destination-zone dmz
   [Device-security-policy-ip-1-loglocalout] source-ip-host 2.2.2.2
   [Device-security-policy-ip-1-loglocalout] destination-ip-host 1.2.3.6
   [Device-security-policy-ip-1-loglocalout] action pass
   [Device-security-policy-ip-1-loglocalout] quit
   ```

   # Configure a rule named **trust-untrust** to all the user to access the Internet.

   ```
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-3-trust-untrust] source-zone trust
   [Device-security-policy-ip-3-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-3-trust-untrust] source-ip-subnet 169.1.1.0 24
   [Device-security-policy-ip-3-trust-untrust] action pass
   [Device-security-policy-ip-3-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure flow log settings:

   # Enable NAT logging for session establishment events, session removal events, and active flows. Set the flow log version to **3.0**.

   ```
   [Device] nat log enable
   [Device] nat log flow-begin
   [Device] nat log flow-end
   [Device] nat log flow-active 10
   [Device] userlog flow export version 3
   ```

   # Specify the log host at 1.2.3.6 as the destination for flow log export, set the UDP port number to 2000, and specify 2.2.2.2 as the source IP address for flow log packets.

   ```
   [Device] userlog flow export host 1.2.3.6 port 2000
   [Device] userlog flow export source-ip 2.2.2.2
   ```

Verifying the configuration

# Display the flow log configuration and statistics.

```
[Device] display userlog export
Flow:
  Export flow log as UDP Packet.
  Version: 3.0
  Source ipv4 address: 2.2.2.2
  Log load balance function: Disabled
  Local time stamp: Disabled
  Number of log hosts: 1
```

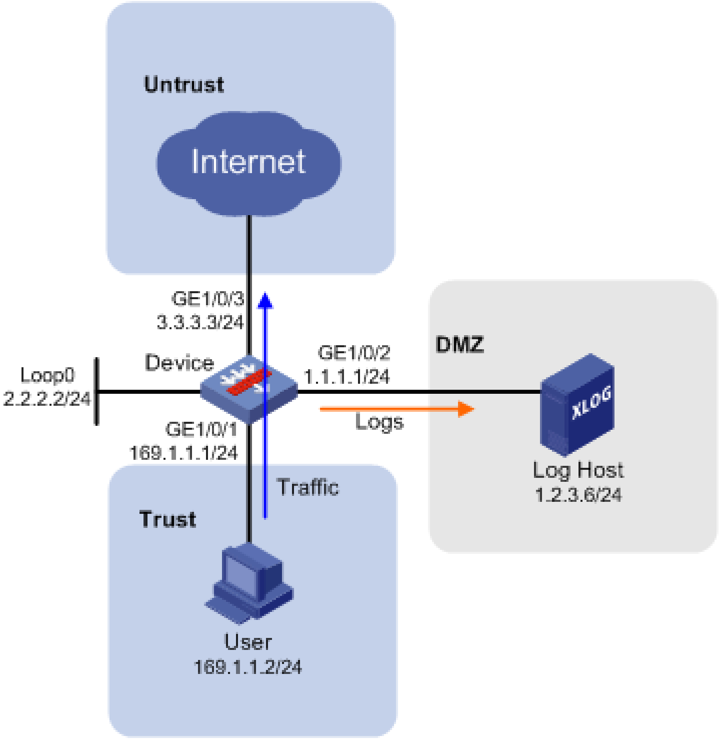
```
Log host 1:
  Host/Port: 1.2.3.6/2000
  Total logs/UDP packets exported: 112/87
```

Example: Configuring session flow log export

Network configuration

As shown in Figure 3, configure flow log on the device to send session flow log entries generated for the user to the log host.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces.

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface loopback 0
   [Device-LoopBack0] ip address 2.2.2.2 255.255.255.0
   [Device-LoopBack0] quit
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] ip address 169.1.1.1 255.255.255.0
   [Device-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures static routes, and the next hops in the routes are 1.1.1.2 and 3.3.3.1, respectively.

   ```
   [Device] ip route-static 1.2.3.0 24 1.1.1.2
   ```

```
[Device] ip route-static 0.0.0.0 0 3.3.3.1
```

3. Add interfaces to security zones.

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface gigabitethernet 1/0/1
[Device-security-zone-Trust] quit
[Device] security-zone name dmz
[Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
[Device-security-zone-DMZ] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-security-zone-Untrust] quit
```

4. Configure a security policy.

# Configure a rule named **loglocalout** to allow the device to send log packets to the log host.

```
[Device] security-policy ip
[Device-security-policy-ip] rule name loglocalout
[Device-security-policy-ip-1-loglocalout] source-zone local
[Device-security-policy-ip-1-loglocalout] destination-zone dmz
[Device-security-policy-ip-1-loglocalout] source-ip-host 2.2.2.2
[Device-security-policy-ip-1-loglocalout] destination-ip-host 1.2.3.6
[Device-security-policy-ip-1-loglocalout] action pass
[Device-security-policy-ip-1-loglocalout] quit
```

# Configure a rule named **trust-untrust** to allow the user to access the Internet.

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-3-trust-untrust] source-zone trust
[Device-security-policy-ip-3-trust-untrust] destination-zone untrust
[Device-security-policy-ip-3-trust-untrust] source-ip-subnet 169.1.1.0 24
[Device-security-policy-ip-3-trust-untrust] action pass
[Device-security-policy-ip-3-trust-untrust] quit
[Device-security-policy-ip] quit
```

5. Configure flow log settings:

# Enable NAT logging for session establishment events and session removal events. Set the flow log version to **3.0**.

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] session log enable ipv4 inbound
[Device] session log flow-begin
[Device] session log flow-end
[Device] userlog flow export version 3
```

# Specify the log host at 1.2.3.6 as the destination for flow log export, set the UDP port number to 2000, and specify 2.2.2.2 as the source IP address for flow log packets.

```
[Device] userlog flow export host 1.2.3.6 port 2000
[Device] userlog flow export source-ip 2.2.2.2
```

Verifying the configuration

# Display the flow log configuration and statistics.

```
[Device] display userlog export
Flow:
  Export flow log as UDP Packet.
  Version: 3.0
```

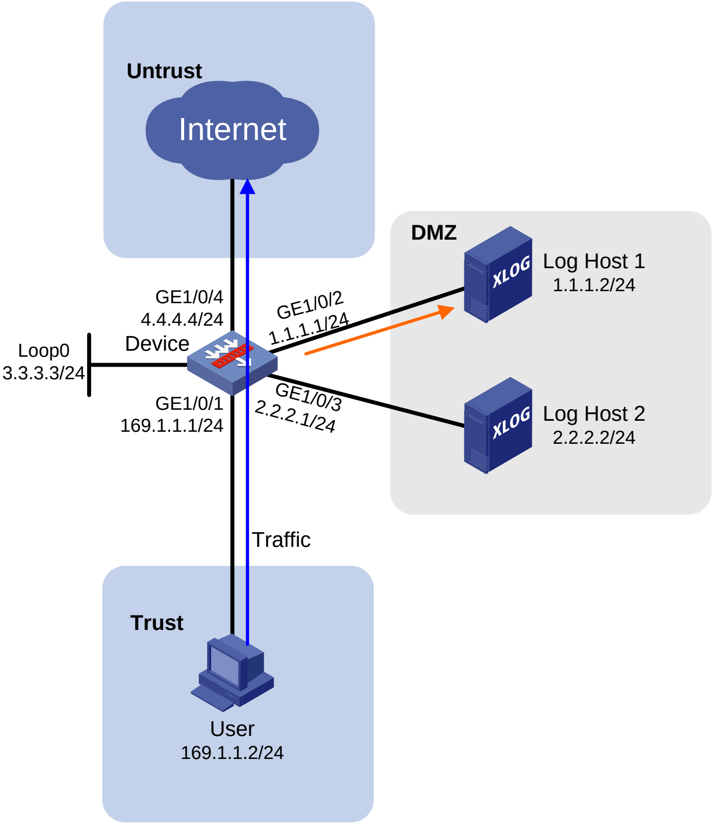
```
Source ipv4 address: 2.2.2.2
Log load balance function: Disabled
Local time stamp: Disabled
Number of log hosts: 1
Log host 1:
  Host/Port: 1.2.3.6/2000
  Total logs/UDP packets exported: 112/87
```

Example: Configuring NAT flow log export to a flow log host group

Network configuration

As shown in Figure 4, configure a flow log host group on the device to send flow log entries generated for the user only to Log Host 1.

Figure 4 Network diagram



Procedure

1. Assign IP addresses to interfaces.

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <Device> system-view
   [Device] interface loopback 0
   [Device-LoopBack0] ip address 3.3.3.3 255.255.255.0
   [Device-LoopBack0] quit
   [Device] interface gigabitethernet 1/0/1
   ```

```
[Device-GigabitEthernet 1/0/1] ip address 169.1.1.1 255.255.255.0
[Device-GigabitEthernet 1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing.

   This example configures a static route, and the next hop in the routes is 4.4.4.1.

   ```
   [Device] ip route-static 0.0.0.0 0 4.4.4.1
   ```

3. Add interfaces to security zones.

   ```
   [Device] security-zone name trust
   [Device-security-zone-Trust] import interface gigabitethernet 1/0/1
   [Device-security-zone-Trust] quit
   [Device] security-zone name dmz
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/2
   [Device-security-zone-DMZ] import interface gigabitethernet 1/0/3
   [Device-security-zone-DMZ] quit
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/4
   [Device-security-zone-Untrust] quit
   ```

4. Configure a security policy.

   # Configure a rule named **loglocalout** for the device to send log packets to the log hosts.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name loglocalout
   [Device-security-policy-ip-1-loglocalout] source-zone local
   [Device-security-policy-ip-1-loglocalout] destination-zone dmz
   [Device-security-policy-ip-1-loglocalout] source-ip-host 3.3.3.3
   [Device-security-policy-ip-1-loglocalout] destination-ip-host 1.1.1.2
   [Device-security-policy-ip-1-loglocalout] destination-ip-host 2.2.2.2
   [Device-security-policy-ip-1-loglocalout] action pass
   [Device-security-policy-ip-1-loglocalout] quit
   ```

   # Configure a rule named **trust-untrust** for the user to access the Internet.

   ```
   [Device-security-policy-ip] rule name trust-untrust
   [Device-security-policy-ip-3-trust-untrust] source-zone trust
   [Device-security-policy-ip-3-trust-untrust] destination-zone untrust
   [Device-security-policy-ip-3-trust-untrust] source-ip-subnet 169.1.1.0 24
   [Device-security-policy-ip-3-trust-untrust] action pass
   [Device-security-policy-ip-3-trust-untrust] quit
   [Device-security-policy-ip] quit
   ```

5. Configure flow log settings.

   # Enable NAT logging for session establishment events, session removal events, and active flows.

   ```
   [Device] nat log enable
   [Device] nat log flow-begin
   [Device] nat log flow-end
   [Device] nat log flow-active 10
   ```

   # Specify the log hosts as the destinations for flow log export, set the UDP port number to 2000. Specify 3.3.3.3 as the source IP address for flow log packets.

   ```
   [Device] userlog flow export host 1.1.1.2 port 2000
   [Device] userlog flow export host 2.2.2.2 port 2000
   [Device] userlog flow export source-ip 3.3.3.3
   ```

# Create ACL 2000 to match packets sent by the user.

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 169.1.1.2 0.0.0.0
[Device-acl-ipv4-basic-2000] quit
```

# Create an IPv4 flow log host group named **test** and specify ACL 2000 for it. Assign Log Host 1 to flow log host group **test**.

```
[Device] userlog host-group test acl number 2000
[Device-userlog-host-group-test] userlog host-group host flow 1.1.1.2
[Device-userlog-host-group-test] quit
```

Verifying the configuration

# Display information about flow log host group **test**.

```
[Device] display userlog host-group test
Userlog host-group test:
  ACL number: 2000

  Flow log host numbers: 1

    Log host 1:
      Host/port: 1.1.1.2/2000
```

# After the user comes online, display flow log export statistics.

```
[Device] display userlog export
Flow:
  Export flow log as UDP Packet.
  Version: 1.0
  Source ipv4 address: 3.3.3.3
  Log load balance function: Disabled
  Local time stamp: Disabled
  Number of log hosts: 2

  Log host 1:
    Host/Port: 1.1.1.2/2000
    Total logs/UDP packets exported: 13/13

  Log host 2:
    Host/Port: 2.2.2.2/2000
    Total logs/UDP packets exported: 0/0
```

Appendix

Appendix A Flow log fields

The fields described in this section are those in the original log data sent to the log host. The log format might differ from the actual log format displayed on the log host. The displayed log format depends on the log analysis tool.

Log header fields

Table 1 shows the log header fields, and the corresponding length and description.

Table 1 Fields in a log header

Field	Length (Bytes)	Description
Version	1	Version number of the log packet: • **1**—Version 1.0. • **3**—Version 3.0. • **5**—Version 5.0.
LogType	1	Log type: • **4**—NAT flow log. • **5**—NAT66 flow log. • **7**—AFT IPv6 to IPv4 log. • **8**—AFT IPv4 to IPv6 log. • **11**—LB NAT64 flow log. • **12**—LB NAT46 flow log. • **13**—LB NAT44 flow log. • **14**—LB NAT66 flow log. • **15**—RIR flow log.
Count	2	Number of flows recorded in the current packet. The value range is 1 to 100.
Second	4	Number of seconds from 1970-01-01 00:00:00 to the time when the packet was generated.
FlowSequence	4	Sequence number of the log packet, which equals to the count of log packets of all log types and versions.
Chassis	2	Number of chassis sent the log packet.
Slot	1	Number of slot sent the log packet.
Cpu	1	Number of CPU sent the log packet.

NAT flow log fields

NAT flow log has three versions: 1.0, 3.0, and 5.0. Table 2, Table 3, and Table 4 show the fields available in the versions.

Table 2 NAT flow log 1.0 fields

Field	Length (Bytes)	Description
SrcIP	4	Source IP address before NAT.

Field	Length (Bytes)	Description
DestIP	4	Destination IP address before NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 if the **Operator** field is 6.
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
Reserved	2	Reserved for future use.

Table 3 NAT flow log 3.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol number.
Operator	1	Reasons why a NAT flow log was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration**.** • **3**—Flow was aged out because of configuration change. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
TosIPv4	1	ToS field of the IPv4 packet.
SourceIP	4	Source IP address before NAT.
SrcNatIP	4	Source IP address after NAT.

Field	Length (Bytes)	Description
DestIP	4	Destination IP address before NAT.
DestNatIP	4	Destination IP address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	1	ID of the source VPN instance.
OutVPNID	1	ID of the destination VPN instance.
vSystemID	2	vSystem ID.
AppID	4	Application protocol ID.
Reserved3	4	Reserved field.

Table 4 NAT flow log 5.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a flow log was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration**.** • **3**—Flow was aged out because of configuration change. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
TosIPv4	1	ToS field of the IPv4 packet.
SourceIP	4	Source IP address before NAT.
SrcNatIP	4	Source IP address after NAT.

Field	Length (Bytes)	Description
DestIP	4	Destination IP address before NAT.
DestNatIP	4	Destination IP address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds.
EndTime	4	End time of the flow, in seconds. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
AppID	4	Application protocol ID.
UserName	56	Username.
vSystemID	2	vSystem ID.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field
Reserved3	4	Reserved field

NAT66 flow log fields

NAT66 flow log has three versions: 1.0, 3.0, and 5.0. Table 5, Table 6, and Table 7 show the fields available in the versions.

Table 5 NAT66 flow log 1.0 fields

Field	Length (Bytes)	Description
SrcIP	16	Source IPv6 address before NAT.
DestIP	16	Destination IPv6 address before NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 if the **Operator** field is 6.
Protocol	1	Protocol type carried by IP.

Field	Length (Bytes)	Description
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
Reserved	2	Reserved for future use.

Table 6 NAT66 flow log 3.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol number.
Operator	1	Reasons why a NAT flow log was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration**.** • **3**—Flow was aged out because of configuration change. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
Traffic class	1	Traffic Class field of the IPv6 packet.
SourceIP	16	Source IPv6 address before NAT.
DestIP	16	Destination IPv6 address before NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.

Field	Length (Bytes)	Description
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
AppID	4	Application protocol ID.

Table 7 NAT66 flow log 5.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a flow log was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration**.** • **3**—Flow was aged out because of configuration change. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
Traffic class	1	Traffic Class field of the IPv6 packet.
SourceIP	16	Source IPv6 address before NAT.
DestIP	16	Destination IPv6 address before NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.

Field	Length (Bytes)	Description
AppID	4	Application protocol ID.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field
Reserved3	4	Reserved field

AFT IPv6 to IPv4 flow log fields

AFT IPv6 to IPv4 flow log has three versions: 1.0, 3.0, and 5.0. Table 8, Table 9, and Table 10 show the fields available in the versions.

Table 8 AFT IPv6 to IPv4 flow log 1.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
Traffic class	1	Traffic Class field of the IPv6 packet.
SourceIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.

Field	Length (Bytes)	Description
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved	4	Reserved field.

Table 9 AFT IPv6 to IPv4 flow log 3.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
Traffic class	1	Traffic Class field of the IPv6 packet.
SourceIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.

Field	Length (Bytes)	Description
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved	4	Reserved field.

Table 10 AFT IPv6 to IPv4 flow log 5.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
Traffic class	1	Traffic Class field of the IPv6 packet.
SourceIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.

Field	Length (Bytes)	Description
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

AFT IPv4 to IPv6 flow log fields

AFT IPv4 to IPv6 flow log has three versions: 1.0, 3.0, and 5.0. Table 11, Table 12, and Table 13 show the fields available in the versions.

Table 11 AFT IPv4 to IPv6 flow log 1.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
TosIPv4	1	ToS field of the IPv4 packet.
SourceIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.

Field	Length (Bytes)	Description
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved	4	Reserved field.

Table 12 AFT IPv4 to IPv6 flow log 3.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
TosIPv4	1	ToS field of the IPv4 packet.
SourceIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1

Field	Length (Bytes)	Description
		0:0.
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved	4	Reserved field.

Table 13 AFT IPv4 to IPv6 flow log 5.0 fields

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
Operator	1	Reasons why a NAT flow log entry was generated: • **0**—Reserved. • **1**—Flow was ended normally. • **2**—Flow was aged out because of aging timer expiration. • **3**—Flow was aged out because of configuration change or manual deletion. • **4**—Flow was aged out because of insufficient resources. • **5**—Reserved. • **6**—Regular connectivity check record for the active flow. • **7**—Flow was deleted because a new flow was created when the flow table was full. • **8**—Flow was created. • **FE**—Other reasons. • **10-FE-1**—Reserved for future use.
IPVersion	1	IP packet version.
TosIPv4	1	ToS field of the IPv4 packet.
SourceIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
StartTime	4	Start time of the flow, in seconds. The value is counted from 1970/1/1 0:0.

Field	Length (Bytes)	Description
EndTime	4	End time of the flow, in seconds. The value is counted from 1970/1/1 0:0. This field is 0 when the **Operator** field is 6.
InTotalPkg	4	Number of packets received for the session.
InTotalByte	4	Number of bytes received for the session.
OutTotalPkg	4	Number of packets sent for the session.
OutTotalByte	4	Number of bytes sent for the session.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

LB NAT64 flow log fields

LB NAT64 flow log has three versions: 1.0, 3.0, and 5.0. Table 14, Table 15, and Table 16 show the fields available in the versions.

Table 14 LB NAT64 flow log 1.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 15 LB NAT64 flow log 3.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 16 LB NAT64 flow log 5.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.

Field	Length (Bytes)	Description
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

LB NAT46 flow log fields

LB NAT46 flow log has three versions: 1.0, 3.0, and 5.0. Table 17, Table 18, and Table 19 show the fields available in the versions.

Table 17 LB NAT46 flow log 1.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 18 LB NAT46 flow log 3.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.

Field	Length (Bytes)	Description
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 19 LB NAT46 flow log 5.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

LB NAT44 flow log fields

LB NAT44 flow log has three versions: 1.0, 3.0, and 5.0. Table 20, Table 21, and Table 22 show the fields available in the versions.

Table 20 LB NAT44 flow log 1.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.

Field	Length (Bytes)	Description
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 21 LB NAT44 flow log 3.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 22 LB NAT44 flow log 5.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	4	Source IPv4 address before NAT.
SrcNatIP	4	Source IPv4 address after NAT.
DestIP	4	Destination IPv4 address before NAT.
DestNatIP	4	Destination IPv4 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

LB NAT66 flow log fields

LB NAT66 flow log has three versions: 1.0, 3.0, and 5.0. Table 23, Table 24, and Table 25 show the fields available in the versions.

Table 23 LB NAT66 flow log 1.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.

Field	Length (Bytes)	Description
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 24 LB NAT66 flow log 3.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
Reserved2	4	Reserved field.

Table 25 LB NAT66 flow log 5.0 fields

Field	Length (Bytes)	Description
LBVersion	1	Version number for the LB log.
Protocol	1	Protocol type carried by IP.
IPVersion	1	IP packet version.
Reserved1	1	Reserved field.
SrcIP	16	Source IPv6 address before NAT.
SrcNatIP	16	Source IPv6 address after NAT.
DestIP	16	Destination IPv6 address before NAT.
DestNatIP	16	Destination IPv6 address after NAT.
SrcPort	2	Source TCP/UDP port number before NAT.

Field	Length (Bytes)	Description
SrcNatPort	2	Source TCP/UDP port number after NAT.
DestPort	2	Destination TCP/UDP port number before NAT.
DestNatPort	2	Destination TCP/UDP port number after NAT.
InVPNID	2	ID of the source VPN instance.
OutVPNID	2	ID of the destination VPN instance.
ContextID	4	ID of the context for the session.
UserName	56	Username.
Reserved1	4	Reserved field.
Reserved2	4	Reserved field.
Reserved3	4	Reserved field.

Contents

Configuring fast log output

About fast log output

The fast log output feature enables fast output of logs to log hosts.

Typically, logs generated by a service module are first sent to the information center, which then outputs the logs to the specified destination (such as to log hosts). When fast log output is configured, logs of service modules are sent directly to log hosts instead of to the information center. Compared to outputting logs to the information center, fast log output saves system resources. For more information about the information center, see "Configuring the information center."

Logs are classified into eight severity levels from 0 through 7 in descending order.

Table 1 Log levels

Severity value	Level	Description
0	Emergency	The system is unusable. For example, the system authorization has expired.
1	Alert	Action must be taken immediately. For example, traffic on an interface exceeds the upper limit.
2	Critical	Critical condition. For example, the device temperature exceeds the upper limit, the power module fails, or the fan tray fails.
3	Error	Error condition. For example, the link state changes.
4	Warning	Warning condition. For example, an interface is disconnected, or the memory resources are used up.
5	Notification	Normal but significant condition. For example, a terminal logs in to the device, or the device reboots.
6	Informational	Informational message. For example, a command or a ping operation is executed.
7	Debugging	Debug message.

Log header formats

The log header formats of fast output logs are as follows:

Table 2 Log header formats

Log header types	Format

Standard format	`<PRI> Timestamp AppName %%10 SN:sn VsysId:id`
	Example:
	`<134> Apr 28 15:35:32 2020 NSFOCUS %%10 SN:10056879 VsysId:1`

Customized format	**URL filtering UNICOM format**:
	`PRI Vision HostName Timestamp AppName MsgID HostName Len`
	Example:
	`142 1 100.0.0.1 2020 Apr 28 15:35:43 NSFOCUS NAT444:SessionU 57`
	NAT CMCC format:
	`<PRI> Vision HostName Timestamp AppName ProcID MsgID`
	Example:
	`<142> 1 100.0.0.1 2020 Apr 28 15:35:32 NSFOCUS - NAT444:SessionA`
	NAT UNICOM format:
	`<PRI> Vision HostName Timestamp AppName ProcID MsgID`
	Example:
	`<142> 1 100.0.0.1 2020 Apr 28 15:35:43 NSFOCUS - NAT444:SessionA`
	NAT TELECOM format:
	`<PRI> Vision Timestamp HostName AppName ProcID MsgID`
	Example:
	`<134> 1 2020 Apr 28 15:35:38 100.0.0.1 NSFOCUS - NAT444:sessionbasedA`

Log field description

Table 3 Log field description

Field	Description
PRI	Log type code. • Standard format and NAT UNICOM format: 134. • URL filtering UNICOM format, NAT CMCC format, and NAT TELECOM format: 142.
Timestamp	Records the time when the log was generated. The timestamp is in the format of YYYY Mon DD hh:mm:ss.
AppName	Name of the device that generated the log.
%%10	Vendor of the device that generated the log.
SN	Serial number of the device that generated the log. To view the device serial number, see the DEVICE_SERIAL_NUMBE field in the output of the **display device manuinfo** command. This field is available only when the device is configured to carry the serial number in fast output logs by using the **customlog with-sn** command.
VsysId	Virtual system that generated the log.
HostName	Source IPv4 address of the device that generated the log.
MsgID	Log type.
Len	Total length of the log header, in bytes.
ProcID	-

Restrictions and guidelines: fast log output configuration

The device supports outputting logs from service modules to log hosts by using the following methods in descending order of priority:

1. Fast log output.
2. Flow log. For more information about flow log and the service modules supported by flow log, see "Configuring flow log."
3. Information center.

If you configure multiple log output methods for a service module, the service module outputs its logs in the method that has the highest priority.

To output NAT logs to a log host, you must specify the log format required by the log host in the **customlog format** and **customlog host** commands.

Configuring fast output of logs to log hosts

1. Enter system view.

   system-view

2. Enable fast log output.

   customlog format { **aft** | **aft-cmcc** | **aft-telecom** | **aft-unicom** | **attack-defense** | **cntm** | **dns** | **dpi** [**anti-virus** | **audit** | **data-filter** | **file-filter** | **ips** [**sgcc** { **policy-hit** | **signature-update** }] | **netshare** | **reputation** | **sandbox** | **terminal** | **traffic-policy** | **url-filter** [**unicom**]] | **keepalive sgcc** | **lb** [**dns-proxy** | **gslb** | **inbound** | **outbound**] | **nat** { **cmcc** | **telecom** | **unicom** } | **packet-filter** [**sgcc**] | **scd** | **security-policy sgcc** | **session** | **trusted-access** { **csap** | **iam** [**authorization** | **notification**] } }

   By default, fast log output is disabled.

3. Configure fast log output parameters.

   customlog host [**vpn-instance** *vpn-instance-name*] { *hostname* | *ipv4-address* | **ipv6** *ipv6-address* } [**port** *port-number*] **export** { **aft** | **attack-defense** | **cmcc-sessionlog** | **cmcc-userlog** | **dns** | **dpi** [**anti-virus** | **audit** | **ips** | **netshare** | **reputation** | **sandbox** | **traffic-policy** | **url-filter**] * | **keepalive** | **lb** [**dns-proxy** | **gslb** | **inbound** | **outbound**] * | **packet-filter** | **scd** | **security-policy** | **session** | **telecom-sessionlog** | **telecom-userlog** | **unicom-sessionlog** | **unicom-userlog** } *

   By default, no fast log output parameters are configured.

   The value for the *port-number* argument must be the same as the port number configured on the log host. Otherwise, the log host cannot receive logs.

4. (Optional.) Specify the source IP address for fast log output.

   customlog host source *interface-type interface-number*

   By default, the source IP address of fast output logs is the primary IP address of the outgoing interface.

   If this command is configured, the primary IP address of the specified interface is used as the source IP address of fast output logs regardless of the outgoing interface.

   Configure this command when you need to filter logs by source IP address on the log host.

5. (Optional.) Configure the timestamp of fast output logs to show the system time.

   customlog timestamp localtime

   By default, the timestamp of fast output logs shows the Greenwich Mean Time (GMT).

6. (Optional.) Configure the device to carry its serial number in fast output logs.

   customlog with-sn

   By default, the device does not carry its serial number in fast output logs.

   Support for this command depends on the device model. For more information, see the command reference.

Configuring fast log output to use the UTF-8 encoding

About this task

The fast log output module and the log host must use the same character set encoding. If they use different encodings, the log host cannot correctly display Chinese characters in the log messages received from the fast log output module. By default, fast log output uses the GB18030 encoding. You can perform this task to configure fast log output to use the UTF-8 encoding.

Procedure

1. Enter system view.

   system-view

2. Configure fast log output to use UTF-8 encoding.

   customlog character-encoding utf-8

   By default, fast log output uses the GB18030 encoding.

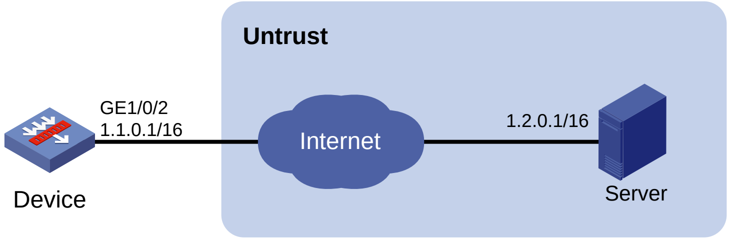
Fast log output configuration examples

Example: Configuring fast log output to a log host

Network configuration

As shown in Figure 1, configure fast log output on the device to send session logs to the log server.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interface GigabitEthernet 1/0/2.

```
<Device> system-view
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] ip address 1.1.0.1 255.255.0.0
```

```
[Device-GigabitEthernet1/0/2] quit
```

2. Configure settings for routing.

   This example configures a static route, and the next hop in the router is 1.1.0.2.

   ```
   [Device] ip route-static 1.2.0.0 16 1.1.0.2
   ```

3. Add interface GigabitEthernet 1/0/2 to security zone **untrust**.

   ```
   [Device] security-zone name untrust
   [Device-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [Device-security-zone-Untrust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **loglocalout** to allow the device to send fast log output messages to the server.

   ```
   [Device] security-policy ip
   [Device-security-policy-ip] rule name loglocalout
   [Device-security-policy-ip-1-loglocalout] source-zone local
   [Device-security-policy-ip-1-loglocalout] destination-zone untrust
   [Device-security-policy-ip-1-loglocalout] source-ip-host 1.1.0.1
   [Device-security-policy-ip-1-loglocalout] destination-ip-host 1.2.0.1
   [Device-security-policy-ip-1-loglocalout] action pass
   [Device-security-policy-ip-1-loglocalout] quit
   [Device-security-policy-ip] quit
   ```

5. Configure fast log output. Enable fast log output, configure log output to the log server, and enable logging for session creation and deletion. Enable IPv4 session logging in the inbound direction of the interface connected to the internal network.

   ```
   [Device] customlog format session
   [Device] customlog host 1.2.0.1 port 1000 export session
   [Device] session log flow-begin
   [Device] session log flow-end
   [Device] interface gigabitethernet 1/0/1
   [Device-GigabitEthernet1/0/1] session log enable ipv4 inbound
   ```

Verifying the configuration

On the server, verify that logs are received from the device successfully.

Contents

Configuring session-based NetStream

About session-based NetStream

Session-based NetStream provides statistics for session-based services and exports the statistics in NetStream v9 format to NetStream servers.

For information about sessions, see session management in *Security Configuration Guide*.

Session-based NetStream aggregation modes

Session-based NetStream aggregates session statistics according to the aggregation criteria of an aggregation mode and exports the statistics to NetStream servers.

Table 1 lists the available aggregation modes. In each mode, the system merges statistics for multiple sessions if each aggregation criterion is of the same value.

Table 1 Session-based NetStream aggregation modes

Aggregation mode	Aggregation criteria
App aggregation	Application layer protocol ID.
App-profile aggregation	• Application layer protocol ID. • Traffic rule ID.
App-user aggregation	• Application layer protocol ID. • User IP address.

Session-based NetStream data export

Session-based NetStream uses an aging mechanism to export flow entry statistics to NetStream servers.

When the aging timer for a session-based NetStream entry expires, statistics about the entry is cleared from the cache and exported to the NetStream servers.

When the session-based NetStream cache is full, the device stops generating new flow entries. Statistics collection for existing flow entries is not affected.

A session-based NetStream entry is also exported in the following situations:

- The session itself ages out.
- The session is manually deleted by the administrator.

Restrictions and guidelines: Session-based NetStream configuration

For session-based NetStream to work, DPI must be enabled on the device. For more information about DPI, see *DPI Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable session-based NetStream.

   session-based netstream enable

   By default, session-based NetStream is disabled.

3. Enable session-based NetStream aggregation modes.

   session-based netstream aggregation { **app** | **app-profile** | **app-user** } *

   By default, all session-based NetStream aggregation modes are disabled.

4. Specify a destination host for session-based NetStream data export.

   session-based netstream export host *ip-address udp-port* [**vpn-instance** *vpn-instance-name*]

   By default, no destination host is specified for session-based NetStream data export.

5. (Optional.) Specify a source IP address for session-based NetStream packets.

   session-based netstream export source ip *ip-address*

   By default, the source IP address of session-based NetStream packets is the primary IP address of the output interface.

6. (Optional.) Set the aging timer for cached session-based NetStream entries.

   session-based netstream timeout *minutes*

   By default, a session-based NetStream entry is can be cached for 5 minutes before being aged out.

Display and maintenance commands session-based NetStream

Execute **display** commands in any view.

Task	Command
Display session-based NetStream statistics in the cache.	**display session-based netstream aggregation-cache** { **app** | **app-profile** | **app-user** } *

Contents

Configuring cloud connections

About cloud connections

A cloud connection is a management tunnel established between a local device and the cloud server. It enables you to manage the local device from the cloud server without accessing the network where the device resides.

Multiple subconnections

After a local device establishes a connection with the cloud server, service modules on the local device can establish multiple subconnections with the microservices on the cloud server. These subconnections are independent from each other and provide separate communication channels for different services. This mechanism avoids interference among different services.

Cloud connection establishment

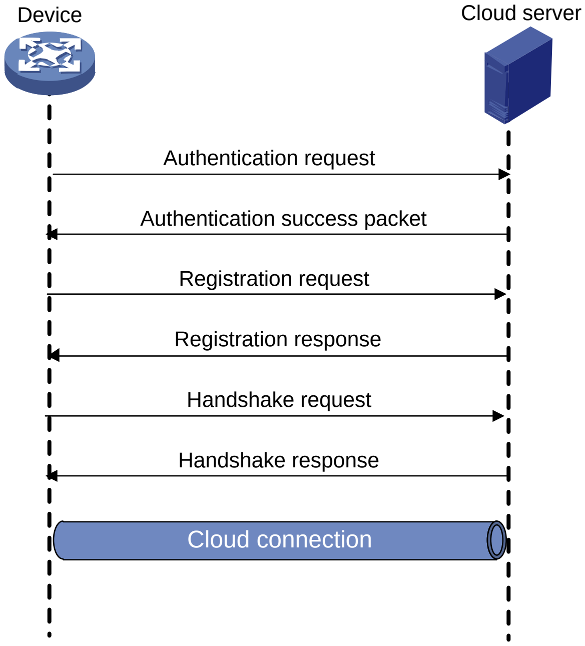
As shown in Figure 1, the cloud connection between the device and the cloud server is established as follows:

1. The device sends an authentication request to the cloud server.
2. The cloud server sends an authentication success packet to the device.

   The device passes the authentication only if the serial number of the device has been added to the cloud server. If the authentication fails, the cloud server sends an authentication failure packet to the device.
3. The device sends a registration request to the cloud server.
4. The cloud server sends a registration response to the device.

   The registration response contains the uniform resource locator (URL) used to establish a cloud connection.
5. The device uses the URL to send a handshake request (changing the protocol from HTTP to WebSocket) to the cloud server.
6. The cloud server sends a handshake response to the device to finish establishing the cloud connection.
7. After the cloud connection is established, the device automatically obtains the subconnection URLs and establishes subconnections with the cloud server based on the service needs.

Figure 1 Establishing a cloud connection



Restrictions: Hardware compatibility with cloud connections

Model	Cloud connection compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Configuring the cloud server

For a successful cloud connection establishment, add the serial number of the device to be managed to the cloud server.

Configuring the local device

About this task

You can specify a cloud server by its domain name and log in to the server through the domain name on a remote PC to manage the local device.

If the local device does not receive a response from the cloud server within three keepalive intervals, the device sends a registration request to re-establish the cloud connection.

To prevent NAT entry aging, the local device sends ping packets to the cloud server periodically.

Restrictions and guidelines

You can specify one primary server by using the **cloud-management server domain** command and a maximum of eight backup servers by repeating the **cloud-management backup-server domain** command.

When establishing a cloud connection, the device connects to one of the primary and backup servers according to the sequence in which they are specified. The first specified server has the highest priority. When the connected server fails, the device switches to another server and does not switch back to the original server even if the original server recovers. To view the connected server, execute the **display cloud-management state** command.

Reduce the ping interval value if the network condition is poor or the NAT entry aging time is short. When you use the ADWAN server for cloud connections, you must set the password for establishing cloud connections to the ADWAN server.

Prerequisites

Before configuring this feature, make sure a DNS server is configured to translate domain names.

Procedure

1. Enter system view.

   system-view

2. Specify the primary cloud server by its domain name.

   cloud-management server domain *domain-name* [**vpn-instance** *vpn-instance-name*] [**source-ip** *ipv4-address*]

3. (Optional.) Specify a backup cloud server by its domain name.

   cloud-management backup-server domain *domain-name* [**vpn-instance** *vpn-instance-name*] [**source-ip** *ipv4-address*]

   By default, no backup cloud server is specified.

4. (Optional.) Set the keepalive interval.

   cloud-management keepalive *interval*

   By default, the keepalive interval is 180 seconds.

5. (Optional.) Set the ping interval.

   cloud-management ping *interval*

   By default, the ping interval is 60 seconds.

6. (Optional.) Specify the TCP port number used to establish cloud connections.

   cloud-management server port *port-number*

   By default, TCP port number 19443 is used to establish cloud connections.

7. (Optional.) Set the password for establishing cloud connections to the ADWAN server.

   cloud-management server password { **cipher** | **simple** } *string*

   By default, no password is set for establishing cloud connections to the ADWAN server.

Unbinding the device from the cloud server

About this task

A device can be registered on the cloud server by only one user.

To register a device that has been registered by another user, you need to take the following steps:

1. Obtain a verification code for device unbinding from the cloud server.
2. Execute the command on the device for sending the verification code to the cloud server.
3. Register the device on the cloud server.

Procedure

1. Enter system view.
   system-view
2. Send the verification code for device unbinding to the cloud server.
   cloud-management unbinding-code *code*

Display and maintenance commands for cloud connections

Execute **display** commands in any view.

Task	Command
Display cloud connection state information.	**display cloud-management state**

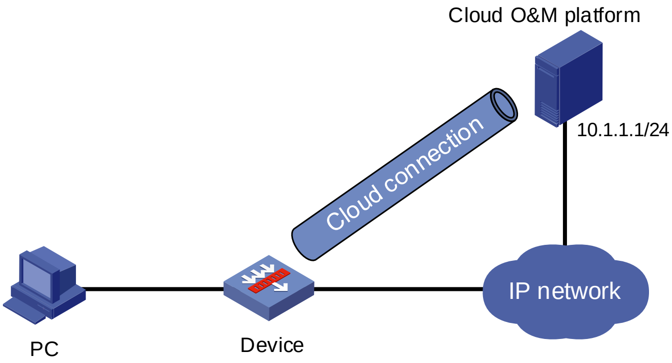
Cloud connection configuration examples

Example: Configuring a cloud connection

Network configuration

As shown in Figure 2, configure the device to establish a cloud connection with the cloud O&M platform.

Figure 2 Network diagram



Procedure

1. Configure IP addresses for interfaces, routes, security domains, and security policies. (Details not shown.)
2. Log in to the cloud O&M platform to add the serial number of the device to the platform. (Details not shown.)
3. Configure the domain name of the cloud O&M platform as **ops.seccloud.**nsfocus**.com** on the device.

```
<Device> system-view
[Device] cloud-management server domain ops.seccloud.nsfocus.com
```

NOTE:

Make sure a DNS server is configured to translate the domain name of the cloud O&M platform.

Verifying the configuration

# Verify that the device and the cloud O&M platform have established a cloud connection.

```
[Device] display cloud-management state
Cloud connection state                       : Established
Device state                                 : Request_success
Cloud server address                         : 10.1.1.1
Cloud server domain name                     : ops.seccloud.nsfocus.com
Cloud connection mode                        : Https
Cloud server port                            : 19443
Connected at                                 : Wed Jan 27 14:18:40 2018
Duration                                     : 00d 00h 02m 01s
Process state                                : Message received
Failure reason                               : N/A
Last down reason                             : socket connection error (Details:N/A)
Last down at                                 : Wed Jan 27 13:18:40 2018
Last report failure reason                   : N/A
Last report failure at                       : N/A
Dropped packets after reaching buffer limit : 0
Total dropped packets                        : 1
Last report incomplete reason                : N/A
Last report incomplete at                    : N/A
Buffer full count                            : 0
```

Contents

Configuring port mirroring

About port mirroring

Port mirroring copies the packets passing through a port to a port that connects to a data monitoring device for packet analysis.

Terminology

The following terms are used in port mirroring configuration.

Mirroring source

The mirroring sources can be one or more monitored ports (called source ports).

Packets passing through mirroring sources are copied to a port connecting to a data monitoring device for packet analysis. The copies are called mirrored packets.

Source device

The device where the mirroring sources reside is called a source device.

Mirroring destination

The mirroring destination connects to a data monitoring device and is the destination port (also known as the monitor port) of mirrored packets. Mirrored packets are sent out of the monitor port to the data monitoring device.

A monitor port might receive multiple copies of a packet when it monitors multiple mirroring sources. For example, two copies of a packet are received on Port A when the following conditions exist:

- Port A is monitoring bidirectional traffic of Port B and Port C on the same device.
- The packet travels from Port B to Port C.

Destination device

The device where the monitor port resides is called the destination device.

Mirroring direction

The mirroring direction specifies the direction of the traffic that is copied on a mirroring source.

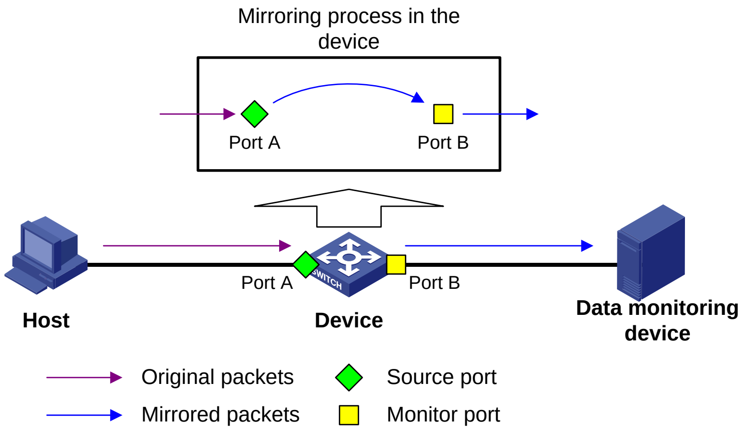
- **Inbound**—Copies packets received.
- **Outbound**—Copies packets sent.
- **Bidirectional**—Copies packets received and sent.

Mirroring group

Port mirroring is implemented through local mirroring groups. The mirroring sources and destination reside on the same device, which is directly connected to a data monitoring device. Packets received on the mirroring sources are sent through the mirroring destination to the data monitoring device.

Local port mirroring

Figure 1 Local port mirroring implementation



As shown in Figure 1, the source port (Port A) and the monitor port (Port B) reside on the same device. Packets received on Port A are copied to Port B. Port B then forwards the packets to the data monitoring device for analysis.

Restrictions: Hardware compatibility with port mirroring

Models	Feature compatibility
NFNX5-HD6480	Yes only on GE 1/0/0 through GE 1/0/13, XGE 1/0/18, XGE 1/0/19, GE 1/0/22 through GE 1/0/29, and interfaces on cards in four interface card slots
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280	Yes only on GE 1/0/0 through GE 1/0/23, XGE 1/0/24, and XGE 1/0/25
NFNX3-HDB1780, NFNX3-HDB3080	Yes only on GE 1/0/0 through GE1/0/23, GE 1/0/25, XGE 1/0/26, and XGE 1/0/27
NFNX3-HDB1180, NFNX3-HDB1480, NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: Port mirroring configuration

You cannot assign a fixed interface and an interface on an interface module to the same mirroring group.

Port mirroring across member devices in an IRF fabric is not supported, which means that the mirroring sources and destination must reside on the same member device.

Port mirroring is not supported on virtual interfaces.

If you configure an interface shared by multiple contexts as a source port, traffic of the source port will be sent to the monitor port without distinction of contexts.

When configuring a mirroring group, first configure the monitor port and then configure the source ports.

When one port mirroring group is configured to monitor the outbound or bidirectional traffic of source ports, the other mirroring group must be configured to monitor the incoming traffic of source ports.

A mirroring group can be configured with one monitor port and multiple source ports. A monitor port or source port of a mirroring group cannot be configured as the source port or monitor port of another mirroring group.

Configuring local port mirroring

Restrictions and guidelines for local port mirroring configuration

A local mirroring group takes effect only after it is configured with the monitor port and mirroring sources.

Local port mirroring tasks at a glance

To configure local port mirroring, perform the following tasks:

1. Creating a local mirroring group
2. Configuring mirroring sources
3. Configuring the monitor port

Creating a local mirroring group

1. Enter system view.
   `system-view`
2. Create a local mirroring group.
   `mirroring-group` *`group-id`* **`local`**

Configuring mirroring sources

Restrictions and guidelines for mirroring source configuration

When you configure source ports for a local mirroring group, follow these restrictions and guidelines:

- A mirroring group can contain multiple source ports.
- A port can act as a source port for multiple mirroring groups.
- A source port cannot be configured as a monitor port.

Configuring source ports

- Configure source ports in system view.
  a. Enter system view.
     `system-view`
  b. Configure source ports for a local mirroring group.

mirroring-group *group-id* **mirroring-port** *interface-list* { **both** | **inbound** | **outbound** }

By default, no source port is configured for a local mirroring group.

- Configure source ports in interface view.
  a. Enter system view.
  system-view
  b. Enter interface view.
  interface *interface-type interface-number*
  c. Configure the port as a source port for a local mirroring group.
  mirroring-group *group-id* **mirroring-port** { **both** | **inbound** | **outbound** }
  By default, a port does not act as a source port for any local mirroring groups.

Configuring the monitor port

Restrictions and guidelines

Do not enable the spanning tree feature on the monitor port.

Only one monitor port can be specified for a local mirroring group.

Use a monitor port only for port mirroring, so the data monitoring device receives only the mirrored traffic.

Procedure

- Configure the monitor port in system view.
  a. Enter system view.
  system-view
  b. Configure the monitor port for a local mirroring group.
  mirroring-group *group-id* **monitor-port** *interface-type interface-number*
  By default, no monitor port is configured for a local mirroring group.
- Configure the monitor port in interface view.
  a. Enter system view.
  system-view
  b. Enter interface view.
  interface *interface-type interface-number*
  c. Configure the port as the monitor port for a mirroring group.
  mirroring-group *group-id* **monitor-port**
  By default, a port does not act as the monitor port for any local mirroring groups.

Display and maintenance commands for port mirroring

Execute **display** commands in any view.

Task	Command
Display mirroring group information.	**display mirroring-group** { *group-id* | **all** | **local** }

Contents

Configuring packet capture

About packet capture

The packet capture feature captures incoming and outgoing packets, generates packet capture records, and saves the records to a .cap file. The file can reside on the device or a remote file server. You can use a packet analyzer such as Wireshark to view the file for traffic analysis.

The minimum packet capture unit is a packet. The packet capture process is as follows:

1. The device captures a specific number of bytes from a packet and generates a packet capture record, ignoring the remaining part of the packet (if any).
2. The device saves the packet capture record in memory.
3. When the maximum number of packet capture records for a file is reached, the device saves the records to a file and clears the records in memory.

Restrictions and guidelines: Packet capture configuration

Start packet capture only when necessary. Packet capture affects device performance.

Only one packet capture process can run on the device.

You can configure packet capture parameters only when packet capture is not started.

If packet capture saves .cap files on the device, back up the .cap files on the device as required after you finish packet capture. Starting packet capture again deletes the existing .cap files.

Packet capture is supported only for the default context and non-default contexts that use their own respective interfaces. It is not supported on shared interfaces of a non-default context. For more information about contexts, see context configuration in *Virtual Technologies Configuration Guide*.

Packet capture tasks at a glance

To configure packet capture, perform the following tasks:

1. Configuring packet capture settings
2. Starting packet capture
3. Stopping packet capture

Configuring packet capture settings

1. Enter system view.

   system-view

2. Set the maximum packet size for a packet capture record.

   packet-capture max-bytes *bytes*

   By default, the maximum packet size is 1600 bytes for a packet capture record.

   To capture all bytes of packets, make sure the maximum packet size for a packet capture record is equal to or greater than the interface MTU.

3. Set the maximum number of packet capture records for a file.

packet-capture max-file-packets *number*

By default, the maximum number of packet capture records is 100 for a file.

4. Specify the storage directory for the .cap files.

   packet-capture storage { **local** [**limit** *limit-space*] | **remote** *serverpath* [**vpn-instance** *vpn-instance-name*] [**user** *username* [**password** { **cipher** | **simple** } *string*]] }

   The default storage directory is the **pcap** directory of the default file system on the master.

Starting packet capture

1. Enter system view.

   system-view

2. Start packet capture.

   packet-capture start [**acl** { *acl-number* | **ipv6** *acl-number* } | **interface** *interface-type interface-number*] *

   By default, the system does not capture packets.

Stopping packet capture

About this task

Saving packet capture records to a file takes time. The **packet-capture stop** command without the **immediately** keyword saves all packet capture records to a file before stopping packet capture. If you do not want to use the packet capture records in memory, execute the **packet-capture stop immediately** command.

Procedure

1. Enter system view.

   system-view

2. Stop packet capture.

   packet-capture stop [**immediately**]

Display and maintenance commands for packet capture

Execute **display** commands in any view.

Task	Command
Display packet capture settings and status information.	**display packet-capture status**

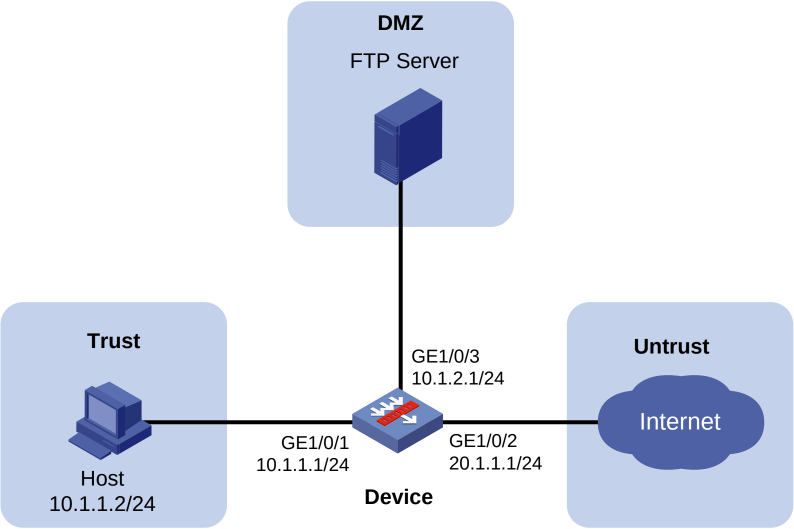
Packet capture configuration examples

Example: Configuring packet capture

Network configuration

As shown in Figure 1, capture packets on GigabitEthernet 1/0/1. Set the maximum packet size for a packet capture record to 3000 bytes. Use a remote file server to save the .cap files.

Figure 1 Network diagram



Procedure

1. Assign IP addresses to interfaces and configure routes, security zones, zone pairs, and interzone policies. Make sure the network connections are available. (Details not shown.)
2. Configure packet capture:

   # Set the storage directory for the .cap files to **ftp://ftp.remote.com/pcap/**. Specify the username and password for accessing the FTP server.

   ```
   [Device] packet-capture storage remote ftp://ftp.remote.com/pcap/ user zhangsan
   password simple 123456TESTplat&!
   ```

   # Set the maximum packet size for a packet capture record to 3000 bytes.

   ```
   [Device] packet-capture max-bytes 3000
   ```

   # Start packet capture on GigabitEthernet 1/0/1.

   ```
   [Device] packet-capture start interface gigabitethernet 1/0/1
   ```

Verifying the configuration

# Display packet capture settings and status information.

```
[Device] display packet-capture status
Capture status: Started
  Filter: Interface GigabitEthernet1/0/1
```

Contents

Configuring NQA

About NQA

Network quality analyzer (NQA) allows you to measure network performance, verify the service levels for IP services and applications, and troubleshoot network problems.

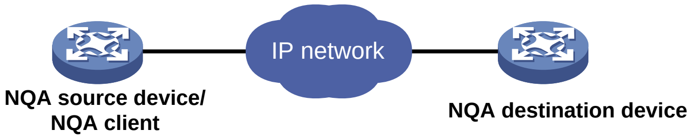
NQA operating mechanism

An NQA operation contains a set of parameters such as the operation type, destination IP address, and port number to define how the operation is performed. Each NQA operation is identified by the combination of the administrator name and the operation tag. You can configure the NQA client to run the operations at scheduled time periods.

As shown in Figure 1, the NQA source device (NQA client) sends data to the NQA destination device by simulating IP services and applications to measure network performance.

All types of NQA operations require the NQA client, but only the TCP, UDP echo, UDP jitter, and voice operations require the NQA server. The NQA operations for services that are already provided by the destination device such as FTP do not need the NQA server. You can configure the NQA server to listen and respond to specific IP addresses and ports to meet various test needs.

Figure 1 Network diagram



After starting an NQA operation, the NQA client periodically performs the operation at the interval specified by using the **`frequency`** command.

You can set the number of probes the NQA client performs in an operation by using the **`probe count`** command.

For the voice and path jitter operations, the NQA client performs only one probe per operation and the **`probe count`** command is not available.

Collaboration with Track

NQA can collaborate with the Track module to notify application modules of state or performance changes so that the application modules can take predefined actions.

The NQA and Track collaboration is available for the following application modules:

- Static routing.
- Policy-based routing.

The following describes how a static route destined for 192.168.0.88 is monitored through collaboration:

1. NQA monitors the reachability to 192.168.0.88.
2. When 192.168.0.88 becomes unreachable, NQA notifies the Track module of the change.
3. The Track module notifies the static routing module of the state change.
4. The static routing module sets the static route to invalid according to a predefined action.

For more information about collaboration, see "Configuring Track."

Threshold monitoring

Threshold monitoring enables the NQA client to take a predefined action when the NQA operation performance metrics violate the specified thresholds.

Table 1 describes the relationships between performance metrics and NQA operation types.

Table 1 Performance metrics and NQA operation types

Performance metric	NQA operation types that can gather the metric
Probe duration	ARP, ICMP echo, DHCP, DNS, FTP, HTTP, SNMP, TCP, UDP echo, and DLSw
Number of probe failures	ARP, ICMP echo, DHCP, DNS, FTP, HTTP, SNMP, TCP, UDP echo, and DLSw
Round-trip time	ICMP jitter, UDP jitter, and voice
Number of discarded packets	ICMP jitter, UDP jitter, and voice
One-way jitter (source-to-destination or destination-to-source)	ICMP jitter, UDP jitter, and voice
One-way delay (source-to-destination or destination-to-source)	ICMP jitter, UDP jitter, and voice
Calculated Planning Impairment Factor (ICPIF) (see "Configuring the voice operation")	Voice
Mean Opinion Scores (MOS) (see "Configuring the voice operation")	Voice

NQA templates

An NQA template is a set of parameters (such as destination address and port number) that defines how an NQA operation is performed. Features such as load balancing and health monitoring can use the NQA template to collect statistics.

You can create multiple NQA templates on the NQA client. Each template must be identified by a unique template name.

Restrictions and guidelines: NQA configuration

To avoid probe failures, follow these restrictions and guidelines when configuring the listening ports on the NQA client and NQA server:

- Do not specify a well-known port.
- Make sure the specified port number is not used by any services on the device.
  - To obtain the IPv4 addresses and the port numbers in use on this device, see the **Local Addr:port** field in the output from the **`display tcp`** and **`display udp`** commands.
  - To obtain the IPv6 addresses and the port numbers in use on this device, see the **LAddr->port** field in the output from the **`display ipv6 tcp`** and **`display ipv6 udp`** commands.

The destination port configured for the operation (with the **`destination port`** command) on the NQA client must be the same as the listening port configured on the server.

NQA tasks at a glance

To configure NQA, perform the following tasks:

1. Configuring the NQA server

   Perform this task on the destination device before you configure the TCP, UDP echo, UDP jitter, and voice operations.

2. Enabling the NQA client
3. Configuring NQA operations or NQA templates

   Choose the following tasks as needed:

   - Configuring NQA operations on the NQA client
   - Configuring NQA templates on the NQA client

   After you configure an NQA operation, you can schedule the NQA client to run the NQA operation.

   An NQA template does not run immediately after it is configured. The template creates and run the NQA operation only when it is required by the feature (such as load balancing) to which the template is applied.

Configuring the NQA server

Restrictions and guidelines

To perform TCP, UDP echo, UDP jitter, and voice operations, you must configure the NQA server on the destination device.

The NQA server listens and responds to requests on the specified IP addresses and ports.

You can configure multiple TCP or UDP listening services on an NQA server, where each corresponds to a specific IP address and port number.

The IP address, port number, and VPN instance for a listening service must be unique on the NQA server and match the configuration on the NQA client.

To perform a UDP jitter operation in high performance mode to the NQA server, enable the high performance mode when configuring the UDP listening service on the NQA server. In this mode, the NQA server cannot process probe packets of the UDP jitter operation if the packet size exceeds 100 bytes and the operation might fail as a result.

Procedure

1. Enter system view.

   system-view

2. Enable the NQA server.

   nqa server enable

   By default, the NQA server is disabled.

3. Configure a TCP listening service.

   nqa server tcp-connect { *ipv4-address* | **ipv6** *ipv6-address* } *port-number* [**vpn-instance** *vpn-instance-name*] [**tos** *tos*]

   This task is required for only TCP and DLSw operations. For the DLSw operation, the port number for the TCP listening service must be 2065.

4. Configure a UDP listening service.

   - Configure a UDP listening service.

     nqa server udp-echo { *ipv4-address* | **ipv6** *ipv6-address* } *port-number* [**vpn-instance** *vpn-instance-name*] [**tos** *tos*]

This task is required for only UDP echo, UDP jitter, and voice operations.

Enabling the NQA client

1. Enter system view.
   `system-view`
2. Enable the NQA client.
   `nqa agent enable`
   By default, the NQA client is enabled.
   The NQA client configuration takes effect after you enable the NQA client.

Configuring NQA operations on the NQA client

NQA operations tasks at a glance

To configure NQA operations, perform the following tasks:

1. Configuring an NQA operation
   - Configuring the ARP operation
   - Configuring the ICMP echo operation
   - Configuring the ICMP jitter operation
   - Configuring the DHCP operation
   - Configuring the DNS operation
   - Configuring the FTP operation
   - Configuring the HTTP operation
   - Configuring the UDP jitter operation
   - Configuring the SNMP operation
   - Configuring the TCP operation
   - Configuring the UDP echo operation
   - Configuring the UDP tracert operation
   - Configuring the voice operation
   - Configuring the DLSw operation
   - Configuring the path jitter operation
2. (Optional.) Configuring optional parameters for the NQA operation
3. (Optional.) Configuring the collaboration feature
4. (Optional.) Configuring threshold monitoring
5. (Optional.) Configuring the NQA statistics collection feature
6. (Optional.) Configuring the saving of NQA history records
7. Scheduling the NQA operation on the NQA client

Configuring the ARP operation

About this task

The ARP operation tests if the ARP service is available on the destination device.

The ARP operation sends an ARP request to the destination device per probe.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the ARP type and enter its view.

   type arp

4. Specify the destination IP address for ARP requests.

   destination ip *ip-address*

   By default, no destination IP address is specified.

5. Specify the source address for ARP requests.

   source ip *ip-address*

   By default, the source IP address of ICMP echo requests is the primary IP address of their output interface.

   The specified source interface must be up.

Configuring the ICMP echo operation

About this task

The ICMP echo operation measures the reachability of a destination device. It has the same function as the **ping** command, but provides more output information. In addition, if multiple paths exist between the source and destination devices, you can specify the next hop for the ICMP echo operation.

The ICMP echo operation sends an ICMP echo request to the destination device per probe.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the ICMP echo type and enter its view.

   type icmp-echo

4. Specify the destination IP address for ICMP echo requests. Choose one option as needed:

   - Specify the destination IP address for ICMP echo requests.

     IPv4:

     destination ip *ip-address*

     IPv6:

     destination ipv6 *ipv6-address*

     By default, no destination IP address is specified.

   - Specify the destination URL for the ICMP echo operation.

     url *url*

     By default, no destination URL is specified for an ICMP echo operation.

     The URL is in the format of *protocol*://*host*:*port, for example, http://host:port.* The *host* parameter is required, and the *protocol* and *port* parameters can be unspecified or be any value.

5. Specify the source address for ICMP echo requests. Choose one option as needed:

- Use the IP address of the specified interface as the source IP address.

  source interface *interface-type interface-number*

  By default, the source IP address of ICMP echo requests is the primary IP address of their output interface.

  The specified source interface must be up.
- Specify the source IPv4 address.

  source ip *ip-address*

  By default, the source IPv4 address of ICMP echo requests is the primary IPv4 address of their output interface.

  The specified source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.
- Specify the source IPv6 address.

  source ipv6 *ipv6-address*

  By default, the source IPv6 address of ICMP echo requests is the IPv6 address of their output interface.

  The specified source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. Specify the output interface or the next hop IP address for ICMP echo requests. Choose one option as needed:
   - Specify the output interface.

     out interface *interface-type interface-number*

     By default, the output interface is not specified. The NQA client determines the output interface based on the routing table lookup.
   - Specify the next hop IPv4 address.

     next-hop ip *ip-address*

     By default, no next hop IPv4 address is specified.
   - Specify the next hop IPv6 address.

     next-hop ipv6 *ipv6-address*

     By default, no next hop IPv6 address is specified.
7. (Optional.) Set the payload size for each ICMP echo request.

   data-size *size*

   The default payload size is 100 bytes.
8. (Optional.) Specify the payload fill string for ICMP echo requests.

   data-fill *string*

   The default payload fill string is the hexadecimal string 00010203040506070809.
9. (Optional.) Enable link connectivity probing for a redundant Ethernet (Reth) member interface.

   reth-member probe enable

   By default, the link connectivity probing is disabeld for Reth member interfaces.

   To probe the link connectivity for two Reth member interfaces with their peers respectively, you can configure two ICMP echo operations, each using one Reth member interface as the output interface

Configuring the ICMP jitter operation

About this task

The ICMP jitter operation measures unidirectional and bidirectional jitters. The operation result helps you to determine whether the network can carry jitter-sensitive services such as real-time voice and video services.

The ICMP jitter operation works as follows:

1. The NQA client sends ICMP packets to the destination device.
2. The destination device time stamps each packet it receives, and then sends the packet back to the NQA client.
3. Upon receiving the responses, the NQA client calculates the jitter according to the timestamps.

The ICMP jitter operation sends a number of ICMP packets to the destination device per probe. The number of packets to send is determined by using the **probe packet-number** command.

Restrictions and guidelines

The **display nqa history** command does not display the results or statistics of the ICMP jitter operation. To view the results or statistics of the operation, use the **display nqa result** or **display nqa statistics** command.

Before starting the operation, make sure the network devices are time synchronized by using NTP. For more information about NTP, see "Configuring NTP."

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the ICMP jitter type and enter its view.

   type icmp-jitter

4. Specify the destination IP address for ICMP packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.

5. Set the number of ICMP packets sent per probe.

   probe packet-number *number*

   The default setting is 10.

6. Set the interval for sending ICMP packets.

   probe packet-interval *interval*

   The default setting is 20 milliseconds.

7. Specify how long the NQA client waits for a response from the server before it regards the response times out.

   probe packet-timeout *timeout*

   The default setting is 3000 milliseconds.

8. Specify the source IP address for ICMP packets.

   source ip *ip-address*

   By default, the source IP address of ICMP packets is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no ICMP packets can be sent out.

Configuring the DHCP operation

About this task

The DHCP operation measures whether or not the DHCP server can respond to client requests. DHCP also measures the amount of time it takes the NQA client to obtain an IP address from a DHCP server.

The NQA client simulates the DHCP relay agent to forward DHCP requests for IP address acquisition from the DHCP server. The interface that performs the DHCP operation does not change its IP address. When the DHCP operation completes, the NQA client sends a packet to release the obtained IP address.

The DHCP operation acquires an IP address from the DHCP server per probe.

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the DHCP type and enter its view.

   `type dhcp`

4. Specify the IP address of the DHCP server as the destination IP address of DHCP packets.

   `destination ip` *`ip-address`*

   By default, no destination IP address is specified.

5. Specify the output interface for DHCP request packets.

   `out interface` *`interface-type interface-number`*

   By default, the NQA client determines the output interface based on the routing table lookup.

6. Specify the source IP address of DHCP request packets.

   `source ip` *`ip-address`*

   By default, the source IP address of DHCP request packets is the primary IP address of their output interface.

   The specified source IP address must be the IP address of a local interface, and the local interface must be up. Otherwise, no probe packets can be sent out.

Configuring the DNS operation

About this task

The DNS operation simulates domain name resolution, and it measures the time for the NQA client to resolve a domain name into an IP address through a DNS server. The obtained DNS entry is not saved.

The DNS operation resolves a domain name into an IP address per probe.

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the DNS type and enter its view.

   `type dns`

4. Specify the IP address of the DNS server as the destination IP address of DNS packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.

5. Specify the domain name to be translated.

   resolve-target *domain-name*

   By default, no domain name is specified.

Configuring the FTP operation

About this task

The FTP operation measures the time for the NQA client to transfer a file to or download a file from an FTP server.

The FTP operation uploads or downloads a file from an FTP server per probe.

Restrictions and guidelines

To upload (**put**) a file to the FTP server, use the **filename** command to specify the name of the file you want to upload. The file must exist on the NQA client.

To download (**get**) a file from the FTP server, include the name of the file you want to download in the **url** command. The file must exist on the FTP server. The NQA client does not save the file obtained from the FTP server.

Use a small file for the FTP operation. A big file might result in transfer failure because of timeout, or might affect other services because of the amount of network bandwidth it occupies.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the FTP type and enter its view.

   type ftp

4. Specify an FTP login username.

   username *username*

   By default, no FTP login username is specified.

5. Specify an FTP login password.

   password { **cipher** | **simple** } *string*

   By default, no FTP login password is specified.

6. Specify the source IP address for FTP request packets.

   source ip *ip-address*

   By default, the source IP address of FTP request packets is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no FTP requests can be sent out.

7. Set the data transmission mode.

   mode { **active** | **passive** }

   The default mode is **active**.

8. Specify the FTP operation type.

operation { **get** | **put** }

The default FTP operation type is **get**.

9. Specify the destination URL for the FTP operation.

   url *url*

   By default, no destination URL is specified for an FTP operation.

   Enter the URL in one of the following formats:

   - ftp://*host/filename*.
   - ftp://*host:port/filename*.

   The *filename* argument is required only for the **get** operation.

10. Specify the name of the file to be uploaded.

    filename *file-name*

    By default, no file is specified.

    This step is required only for the **put** operation.

    The configuration does not take effect for the **get** operation.

Configuring the HTTP operation

About this task

The HTTP operation measures the time for the NQA client to obtain responses from an HTTP server.

The HTTP operation supports the following operation types:

- **Get**—Retrieves data such as a Web page from the HTTP server.
- **Post**—Sends data to the HTTP server for processing.
- **Raw**—Sends a user-defined HTTP request to the HTTP server. You must manually configure the content of the HTTP request to be sent.

The HTTP operation completes the operation of the specified type per probe.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the HTTP type and enter its view.

   type http

4. Specify the destination URL for the HTTP operation.

   url *url*

   By default, no destination URL is specified for an HTTP operation.

   Enter the URL in one of the following formats:

   - http://host/resource
   - http://host:port/resource

5. Specify an HTTP login username.

   username *username*

   By default, no HTTP login username is specified.

6. Specify an HTTP login password.

   password { **cipher** | **simple** } *string*

By default, no HTTP login password is specified.

7. Specify the HTTP version.

   `version` { **`v1.0`** | **`v1.1`** }

   By default, HTTP 1.0 is used.

8. Specify the HTTP operation type.

   `operation` { **`get`** | **`post`** | **`raw`** }

   The default HTTP operation type is **`get`**.

   If you set the operation type to **`raw`**, the client pads the content configured in raw request view to the HTTP request to send to the HTTP server.

9. Configure the HTTP raw request.

   a. Enter raw request view.

      `raw-request`

      Every time you enter raw request view, the previously configured raw request content is cleared.

   b. Enter or paste the request content.

      By default, no request content is configured.

      For successful HTTP operations, make sure the raw request content is valid and does not contain command aliases configured by using the **`alias`** command. For more information about the **`alias`** command, see CLI commands in *Fundamentals Command Reference*.

   c. Save the input and return to HTTP operation view.

      `quit`

   This step is required only when the operation type is set to **`raw`**.

10. Specify the source IP address for the HTTP packets.

    `source ip` *`ip-address`*

    By default, the source IP address of HTTP packets is the primary IP address of their output interface.

    The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no HTTP packets can be sent out.

Configuring the UDP jitter operation

About this task

The UDP jitter operation measures unidirectional and bidirectional jitters. The operation result helps you determine whether the network can carry jitter-sensitive services such as real-time voice and video services.

The UDP jitter operation works as follows:

1. The NQA client sends UDP packets to the destination port.
2. The destination device time stamps each packet it receives, and then sends the packet back to the NQA client.
3. Upon receiving the responses, the NQA client calculates the jitter according to the timestamps.

The UDP jitter operation sends a number of UDP packets to the destination device per probe. The number of packets to send is determined by using the **`probe packet-number`** command.

The UDP jitter operation requires both the NQA server and the NQA client. Before you perform the UDP jitter operation, configure the UDP listening service on the NQA server. For more information about UDP listening service configuration, see "Configuring the NQA server."

Restrictions and guidelines

To ensure successful UDP jitter operations and avoid affecting existing services, do not perform the operations on well-known ports from 1 to 1023.

The **`display nqa history`** command does not display the results or statistics of the UDP jitter operation. To view the results or statistics of the UDP jitter operation, use the **`display nqa result`** or **`display nqa statistics`** command.

Before starting the operation, make sure the network devices are time synchronized by using NTP. For more information about NTP, see "Configuring NTP."

If you want to enable high performance mode for all UDP jitter operations on the device, use the **`nqa agent udp-jitter high-performance enable`** command in system view.

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the UDP jitter type and enter its view.

   `type udp-jitter`

4. Specify the destination IP address and destination port number for UDP packets. Choose one option as needed:

   - Specify the destination IP address and destination port number for UDP jitter requests.

     `destination port` *`port-number`*

     By default, no destination port number is specified.

     The destination port number must be the same as the port number of the UDP listening service on the NQA server.

     IPv4:

     `destination ip` *`ip-address`*

     By default, no destination IPv4 address is specified.

     The destination IPv4 address must be the same as the IP address of the UDP listening service configured on the NQA server.

   IPv6:

   `destination ipv6` *`ipv6-address`*

   By default, no destination IPv6 address is specified.

   The destination IPv6 address must be the same as the IPv6 address of the UDP listening service configured on the NQA server.

   - Specify the destination URL for the UDP jitter operation.

     `url` *`url`*

     By default, no destination URL is specified for a UDP jitter operation.

     The URL is in the format of *protocol://host:port*, for example, http://host:port. The *host* and *port* parameters are required, and the *protocol* parameter can be unspecified or any value.

5. Specify the source IP address for UDP packets.

   IPv4:

   `source ip` *`ip-address`*

   By default, the source IPv4 address of UDP packets is the primary IPv4 address of their output interface.

The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no UDP packets can be sent out.

IPv6:

`source ipv6` *`ipv6-address`*

By default, the source IPv6 address of UDP packets is the IPv6 address of their output interface.

The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no UDP packets can be sent out.

6. Specify the source port number for UDP packets.

   `source port` *`port-number`*

   By default, the NQA client randomly picks an unused port as the source port when the operation starts.

   For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

7. Specify an output interface for UDP packets.

   `out interface` *`interface-type interface-number`*

   By default, the NQA client determines the output interface based on the routing table lookup.

8. Set the number of UDP packets sent per probe.

   `probe packet-number` *`number`*

   The default setting is 10.

9. Set the interval for sending UDP packets.

   `probe packet-interval` *`interval`*

   The default setting is 20 milliseconds.

10. Specify how long the NQA client waits for a response from the server before it regards the response times out.

    `probe packet-timeout` *`timeout`*

    The default setting is 3000 milliseconds.

11. (Optional.) Set the payload size for each UDP packet.

    `data-size` *`size`*

    The default payload size is 100 bytes.

12. (Optional.) Specify the payload fill string for UDP packets.

    `data-fill` *`string`*

    The default payload fill string is the hexadecimal string 00010203040506070809.

Configuring the SNMP operation

About this task

The SNMP operation tests whether the SNMP service is available on an SNMP agent.

The SNMP operation sends one SNMPv1 packet, one SNMPv2c packet, and one SNMPv3 packet to the SNMP agent per probe.

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the SNMP type and enter its view.

   type snmp

4. Specify the destination IP address for SNMP packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.

5. Specify the destination port number for SNMP packets.

   destination port *port-number*

   The default destination port number of SNMP packets is 161.

6. Specify the source IP address for SNMP packets.

   source ip *ip-address*

   By default, the source IP address of SNMP packets is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no SNMP packets can be sent out.

7. Specify the source port number for SNMP packets.

   source port *port-number*

   By default, the NQA client randomly picks an unused port as the source port when the operation starts.

   For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

8. Specify the community name carried in the SNMPv1 and SNMPv2c packets.

   community read { **cipher** | **simple** } *community-name*

   By default, the SNMPv1 and SNMPv2c packets carry community name **public**.

   Make sure the specified community name is the same as the community name configured on the SNMP agent.

Configuring the TCP operation

About this task

The TCP operation measures the time for the NQA client to establish a TCP connection to a port on the NQA server.

The TCP operation requires both the NQA server and the NQA client. Before you perform a TCP operation, configure a TCP listening service on the NQA server. For more information about the TCP listening service configuration, see "Configuring the NQA server."

The TCP operation sets up a TCP connection per probe.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the TCP type and enter its view.

   type tcp

4. Specify the destination address for TCP packets.

   IPv4:

   destination ip *ip-address*

By default, no destination IPv4 address is specified.

The destination IPv4 address must be the same as the IPv4 address of the TCP listening service configured on the NQA server.

IPv6:

destination ipv6 *ipv6-address*

By default, no destination IPv6 address is specified.

The destination IPv6 address must be the same as the IPv6 address of the TCP listening service configured on the NQA server.

5. Specify the destination port number for TCP packets.

   destination port *port-number*

   By default, no destination port number is configured.

   The destination port number must be the same as the port number of the listening service on the NQA server.

6. Specify the source IP address for TCP packets.

   IPv4:

   source ip *ip-address*

   By default, the source IPv4 address of TCP packets is the primary IPv4 address of their output interface.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no TCP packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of TCP packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no TCP packets can be sent out.

Configuring the UDP echo operation

About this task

The UDP echo operation measures the round-trip time between the client and a UDP port on the NQA server.

The UDP echo operation requires both the NQA server and the NQA client. Before you perform a UDP echo operation, configure a UDP listening service on the NQA server. For more information about the UDP listening service configuration, see "Configuring the NQA server."

The UDP echo operation sends a UDP packet to the destination device per probe.

Procedure

1. Enter system view.

   system-view

2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*

3. Specify the UDP echo type and enter its view.

   type udp-echo

4. Specify the destination address for UDP packets.

   IPv4:

   destination ip *ip-address*

   By default, no destination IPv4 address is specified.

The destination IPv4 address must be the same as the IPv4 address of the UDP listening service configured on the NQA server.

IPv6:

destination ipv6 *ipv6-address*

By default, no destination IPv6 address is specified.

The destination IPv6 address must be the same as the IPv6 address of the UDP listening service configured on the NQA server.

5. Specify the destination port number for UDP packets.

   destination port *port-number*

   By default, no destination port number is specified.

   The destination port number must be the same as the port number of the listening service on the NQA server.

6. Specify the source IP address for UDP packets.

   IPv4:

   source ip *ip-address*

   By default, the source IPv4 address of UDP packets is the primary IPv4 address of their output interface.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no UDP packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of UDP packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no UDP packets can be sent out.

7. Specify the source port number for UDP packets.

   source port *port-number*

   By default, the NQA client randomly picks an unused port as the source port when the operation starts.

   For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

8. (Optional.) Set the payload size for each UDP packet.

   data-size *size*

   The default setting is 100 bytes.

9. (Optional.) Specify the payload fill string for UDP packets.

   data-fill *string*

   The default payload fill string is the hexadecimal string 00010203040506070809.

Configuring the UDP tracert operation

About this task

The UDP tracert operation determines the routing path from the source device to the destination device.

The UDP tracert operation sends a UDP packet to a hop along the path per probe.

Restrictions and guidelines

The UDP tracert operation is not supported on IPv6 networks. To determine the routing path that the IPv6 packets traverse from the source to the destination, use the **tracert ipv6** command. For more information about the command, see *Network Management and Monitoring Command Reference.*

Prerequisites

Before you configure the UDP tracert operation, you must perform the following tasks:

- Enable sending ICMP time exceeded messages on the intermediate devices between the source and destination devices. If the intermediate devices are NSFOCUS devices, use the **ip ttl-expires enable** command.
- Enable sending ICMP destination unreachable messages on the destination device. If the destination device is an NSFOCUS device, use the **ip unreachables enable** command.

  For more information about the **ip ttl-expires enable** and **ip unreachables enable** commands, see *Layer 3—IP Services Command Reference*.

Procedure

1. Enter system view.

   system-view
2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*
3. Specify the UDP tracert operation type and enter its view.

   type udp-tracert
4. Specify the destination IP address for the UDP packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.
5. Specify the destination port number for the UDP packets.

   destination port *port-number*

   By default, the destination port number is 33434.

   The specified port number must be an unused port number on the destination device, so the destination device can reply with an ICMP port unreachable message.
6. Specify an output interface for UDP packets.

   out interface *interface-type interface-number*

   By default, the NQA client determines the output interface based on the routing table lookup.
7. Specify the source IP address for UDP packets.
   - Specify the IP address of the specified interface as the source IP address.

     source interface *interface-type interface-number*

     By default, the source IP address of UDP packets is the primary IP address of their output interface.
   - Specify the source IP address.

     source ip *ip-address*

     The specified source interface must be up. The source IP address must be the IP address of a local interface, and the local interface must be up. Otherwise, no probe packets can be sent out.
8. Specify the source port number for UDP packets.

   source port *port-number*

By default, the NQA client randomly picks an unused port as the source port when the operation starts.

For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

9. Set the maximum number of consecutive probe failures.

   `max-failure` *`times`*

   The default setting is 5.

10. Set the initial TTL value for UDP packets.

    `init-ttl` *`value`*

    The default setting is 1.

11. (Optional.) Set the payload size for each UDP packet.

    `data-size` *`size`*

    The default setting is 100 bytes.

12. (Optional.) Enable the no-fragmentation feature.

    `no-fragment enable`

    By default, the no-fragmentation feature is disabled.

Configuring the voice operation

About this task

The voice operation measures VoIP network performance.

The voice operation works as follows:

1. The NQA client sends voice packets at sending intervals to the destination device (NQA server).

   The voice packets are of one of the following codec types:

   - G.711 A-law.
   - G.711 μ-law.
   - G.729 A-law.

2. The destination device time stamps each voice packet it receives and sends it back to the source.
3. Upon receiving the packet, the source device calculates the jitter and one-way delay based on the timestamp.

The voice operation sends a number of voice packets to the destination device per probe. The number of packets to send per probe is determined by using the **`probe packet-number`** command.

The following parameters that reflect VoIP network performance can be calculated by using the metrics gathered by the voice operation:

- **Calculated Planning Impairment Factor (ICPIF)**—Measures impairment to voice quality on a VoIP network. It is decided by packet loss and delay. A higher value represents a lower service quality.
- **Mean Opinion Scores (MOS)**—A MOS value can be evaluated by using the ICPIF value, in the range of 1 to 5. A higher value represents a higher service quality.

The evaluation of voice quality depends on users' tolerance for voice quality. For users with higher tolerance for voice quality, use the **`advantage-factor`** command to set an advantage factor. When the system calculates the ICPIF value, it subtracts the advantage factor to modify ICPIF and MOS values for voice quality evaluation.

The voice operation requires both the NQA server and the NQA client. Before you perform a voice operation, configure a UDP listening service on the NQA server. For more information about UDP listening service configuration, see "Configuring the NQA server."

Restrictions and guidelines

To ensure successful voice operations and avoid affecting existing services, do not perform the operations on well-known ports from 1 to 1023.

The **`display nqa history`** command does not display the results or statistics of the voice operation. To view the results or statistics of the voice operation, use the **`display nqa result`** or **`display nqa statistics`** command.

Before starting the operation, make sure the network devices are time synchronized by using NTP. For more information about NTP, see "Configuring NTP."

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the voice type and enter its view.

   `type voice`

4. Specify the destination IP address for voice packets.

   `destination ip` *`ip-address`*

   By default, no destination IP address is configured.

   The destination IP address must be the same as the IP address of the UDP listening service on the NQA server.

5. Specify the destination port number for voice packets.

   `destination port` *`port-number`*

   By default, no destination port number is configured.

   The destination port number must be the same as the port number of the listening service on the NQA server.

6. Specify the source IP address for voice packets.

   `source ip` *`ip-address`*

   By default, the source IP address of voice packets is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no voice packets can be sent out.

7. Specify the source port number for voice packets.

   `source port` *`port-number`*

   By default, the NQA client randomly picks an unused port as the source port when the operation starts.

   For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

8. Configure the basic voice operation parameters.

   - Specify the codec type.

     `codec-type` `{` **`g711a`** `|` **`g711u`** `|` **`g729a`** `}`

     By default, the codec type is G.711 A-law.

   - Set the advantage factor for calculating MOS and ICPIF values.

     `advantage-factor` *`factor`*

By default, the advantage factor is 0.

9. Configure the probe parameters for the voice operation.
   - Set the number of voice packets to be sent per probe.

     probe packet-number *number*

     The default setting is 1000.
   - Set the interval for sending voice packets.

     probe packet-interval *interval*

     The default setting is 20 milliseconds.
   - Specify how long the NQA client waits for a response from the server before it regards the response times out.

     probe packet-timeout *timeout*

     The default setting is 5000 milliseconds.
10. Configure the payload parameters.
    a. Set the payload size.

     data-size *size*

     For the G.711A-law and G.711 µ-law codec types, the default payload size is 172 bytes.

     For the G.729 A-law codec type, the default payload size is 32 bytes.

    b. (Optional.) Specify the payload fill string for voice packets.

     data-fill *string*

     The default payload fill string is the hexadecimal string 00010203040506070809.

Configuring the DLSw operation

About this task

The DLSw operation measures the response time of a DLSw device.

It sets up a DLSw connection to the DLSw device per probe.

Restrictions and guidelines

For the successful DLSw operation, configure the **nqa server tcp-connect** command on the NQA server and make sure the port number for the TCP listening service is 2065.

Procedure

1. Enter system view.

   system-view
2. Create an NQA operation and enter NQA operation view.

   nqa entry *admin-name operation-tag*
3. Specify the DLSw type and enter its view.

   type dlsw
4. Specify the destination IP address for the probe packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.
5. Specify the source IP address for the probe packets.

   source ip *ip-address*

   By default, the source IP address of the probe packets is the primary IP address of their output interface.

The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the path jitter operation

About this task

The path jitter operation measures the jitter, negative jitters, and positive jitters from the NQA client to each hop on the path to the destination.

The path jitter operation performs the following steps per probe:

1. Obtains the path from the NQA client to the destination through tracert. A maximum of 64 hops can be detected.
2. Sends a number of ICMP echo requests to each hop along the path. The number of ICMP echo requests to send is set by using the **`probe packet-number`** command.

Prerequisites

Before you configure the path jitter operation, you must perform the following tasks:

- Enable sending ICMP time exceeded messages on the intermediate devices between the source and destination devices. If the intermediate devices are NSFOCUS devices, use the **`ip ttl-expires enable`** command.
- Enable sending ICMP destination unreachable messages on the destination device. If the destination device is an NSFOCUS device, use the **`ip unreachables enable`** command.

  For more information about the **`ip ttl-expires enable`** and **`ip unreachables enable`** commands, see *Layer 3—IP Services Command Reference*.

Procedure

1. Enter system view.

   `system-view`

2. Create an NQA operation and enter NQA operation view.

   `nqa entry` *`admin-name operation-tag`*

3. Specify the path jitter type and enter its view.

   `type path-jitter`

4. Specify the destination IP address for ICMP echo requests.

   `destination ip` *`ip-address`*

   By default, no destination IP address is specified.

5. Specify the source IP address for ICMP echo requests.

   `source ip` *`ip-address`*

   By default, the source IP address of ICMP echo requests is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no ICMP echo requests can be sent out.

6. Configure the probe parameters for the path jitter operation.

   a. Set the number of ICMP echo requests to be sent per probe.

      `probe packet-number` *`number`*

      The default setting is 10.

   b. Set the interval for sending ICMP echo requests.

      `probe packet-interval` *`interval`*

      The default setting is 20 milliseconds.

c. Specify how long the NQA client waits for a response from the server before it regards the response times out.

   probe packet-timeout *timeout*

   The default setting is 3000 milliseconds.

7. (Optional.) Specify an LSR path.

   lsr-path *ip-address*&<1-8>

   By default, no LSR path is specified.

   The path jitter operation uses tracert to detect the LSR path to the destination, and sends ICMP echo requests to each hop on the LSR path.

8. Configure the NQA client to perform the path jitter operation only on the destination address.

   target-only

   By default, the path jitter operation is performed on each hop on the path to the destination.

9. (Optional.) Set the payload size for each ICMP echo request.

   data-size *size*

   The default setting is 100 bytes.

10. (Optional.) Specify the payload fill string for ICMP echo requests.

    data-fill *string*

    The default payload fill string is the hexadecimal string 00010203040506070809.

Configuring optional parameters for the NQA operation

Restrictions and guidelines

The parameter settings take effect only on the current operation.

The supported parameters vary by NQA operation type. For information about supported parameters, see NQA commands in *Network Management and Monitoring Command Reference*.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA operation.

   nqa entry *admin-name operation-tag*

3. Specify an NQA operation type and enter its view.

   type { **arp** | **dhcp** | **dlsw** | **dns** | **ftp** | **http** | **icmp-echo** | **icmp-jitter** | **path-jitter** | **snmp** | **tcp** | **udp-echo** | **udp-jitter** | **udp-tracert** | **voice** }

4. Configure a description for the operation.

   description *text*

   By default, no description is configured.

5. Set the interval at which the NQA operation repeats.

   frequency *interval*

   For a voice or path jitter operation, the default setting is 60000 milliseconds.

   For other types of operations, the default setting is 0 milliseconds, and only one operation is performed.

   When the interval expires, but the operation is not completed or is not timed out, the next operation does not start.

6. Specify the probe times.

   probe count *times*

In an UDP tracert operation, the NQA client performs three probes to each hop to the destination by default.

In other types of operations, the NQA client performs one probe to the destination per operation by default.

This command is not available for the voice and path jitter operations. Each of these operations performs only one probe.

7. Set the probe timeout time.

   probe timeout *timeout*

   The default setting is 3000 milliseconds.

8. Set the maximum number of hops that the probe packets can traverse.

   ttl *value*

   The default setting is 30 for probe packets of the UDP tracert operation, and is 20 for probe packets of other types of operations.

9. Set the ToS value in the IP header of the probe packets.

   tos *value*

   The default setting is 0.

10. Enable the routing table bypass feature.

    route-option bypass-route

    By default, the routing table bypass feature is disabled.

    This command does not take effect if the destination address of the NQA operation is an IPv6 address.

11. Specify the VPN instance where the operation is performed.

    vpn-instance *vpn-instance-name*

    By default, the operation is performed on the public network.

Configuring the collaboration feature

About this task

Collaboration is implemented by associating a reaction entry of an NQA operation with a track entry. The reaction entry monitors the NQA operation. If the number of operation failures reaches the specified threshold, the configured action is triggered.

Restrictions and guidelines

The collaboration feature is not available for the following types of operations:

- ICMP jitter operation.
- UDP jitter operation.
- UDP tracert operation.
- Voice operation.
- Path jitter operation.

A reaction entry cannot be modified after it is created.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA operation.

   nqa entry *admin-name operation-tag*

3. Configure a reaction entry.

reaction *item-number* **checked-element probe-fail threshold-type consecutive** *consecutive-occurrences* **action-type trigger-only**

You cannot modify the content of an existing reaction entry.

4. Return to system view.

   quit

5. Associate Track with NQA.

   For information about the configuration, see "Configuring Track."

6. Associate Track with an application module.

   For information about the configuration, see "Configuring Track."

Configuring threshold monitoring

About this task

This feature allows you to monitor the NQA operation running status.

An NQA operation supports the following threshold types:

- **average**—If the average value for the monitored performance metric either exceeds the upper threshold or goes below the lower threshold, a threshold violation occurs.
- **accumulate**—If the total number of times that the monitored performance metric is out of the specified value range reaches or exceeds the specified threshold, a threshold violation occurs.
- **consecutive**—If the number of consecutive times that the monitored performance metric is out of the specified value range reaches or exceeds the specified threshold, a threshold violation occurs.

Threshold violations for the average or accumulate threshold type are determined on a per NQA operation basis. The threshold violations for the consecutive type are determined from the time the NQA operation starts.

The following actions might be triggered:

- **none**—NQA displays results only on the terminal screen. It does not send traps to the NMS.
- **trap-only**—NQA displays results on the terminal screen, and meanwhile it sends traps to the NMS.

  To send traps to the NMS, the NMS address must be specified by using the **snmp-agent target-host** command. For more information about the command, see *Network Management and Monitoring Command Reference*.
- **trigger-only**—NQA displays results on the terminal screen, and meanwhile triggers other modules for collaboration.

In a reaction entry, configure a monitored element, a threshold type, and an action to be triggered to implement threshold monitoring.

The state of a reaction entry can be invalid, over-threshold, or below-threshold.

- Before an NQA operation starts, the reaction entry is in invalid state.
- If the threshold is violated, the state of the entry is set to over-threshold. Otherwise, the state of the entry is set to below-threshold.

Restrictions and guidelines

The threshold monitoring feature is not available for the path jitter operation.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA operation.

nqa entry *admin-name operation-tag*

3. Enable sending traps to the NMS when specific conditions are met.

   reaction trap { **path-change** | **probe-failure** *consecutive-probe-failures* | **test-complete** | **test-failure** [*accumulate-probe-failures*] }

   By default, no traps are sent to the NMS.

   The ICMP jitter, UDP jitter, and voice operations support only the **test-complete** keyword.

   The following parameters are not available for the UDP tracert operation:

   - The **probe-failure** *consecutive-probe-failures* option.
   - The *accumulate-probe-failures* argument.

4. Configure threshold monitoring. Choose the options to configure as needed:

   - Monitor the operation duration.

     reaction *item-number* **checked-element probe-duration threshold-type** { **accumulate** *accumulate-occurrences* | **average** | **consecutive** *consecutive-occurrences* } **threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     This reaction entry is not supported in the ICMP jitter, UDP jitter, UDP tracert, or voice operations.

   - Monitor failure times.

     reaction *item-number* **checked-element probe-fail threshold-type** { **accumulate** *accumulate-occurrences* | **consecutive** *consecutive-occurrences* } [**action-type** { **none** | **trap-only** }]

     This reaction entry is not supported in the ICMP jitter, UDP jitter, UDP tracert, or voice operations.

   - Monitor the round-trip time.

     reaction *item-number* **checked-element rtt threshold-type** { **accumulate** *accumulate-occurrences* | **average** } **threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     Only the ICMP jitter, UDP jitter, and voice operations support this reaction entry.

   - Monitor packet loss.

     reaction *item-number* **checked-element packet-loss threshold-type accumulate** *accumulate-occurrences* [**action-type** { **none** | **trap-only** }]

     Only the ICMP jitter, UDP jitter, and voice operations support this reaction entry.

   - Monitor the one-way jitter.

     reaction *item-number* **checked-element** { **jitter-ds** | **jitter-sd** } **threshold-type** { **accumulate** *accumulate-occurrences* | **average** } **threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     Only the ICMP jitter, UDP jitter, and voice operations support this reaction entry.

   - Monitor the one-way delay.

     reaction *item-number* **checked-element** { **owd-ds** | **owd-sd** } **threshold-value** *upper-threshold lower-threshold*

     Only the ICMP jitter, UDP jitter, and voice operations support this reaction entry.

   - Monitor the ICPIF value.

     reaction *item-number* **checked-element icpif threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     Only the voice operation supports this reaction entry.

- Monitor the MOS value.

  reaction *item-number* **checked-element mos threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

  Only the voice operation supports this reaction entry.

The DNS operation does not support the action of sending trap messages. For the DNS operation, the action type can only be **none**.

Configuring the NQA statistics collection feature

About this task

NQA forms statistics within the same collection interval as a statistics group. To display information about the statistics groups, use the **display nqa statistics** command.

When the maximum number of statistics groups is reached, the NQA client deletes the oldest statistics group to save a new one.

A statistics group is automatically deleted when its hold time expires.

Restrictions and guidelines

The NQA statistics collection feature is not available for the UDP tracert operation.

If you use the **frequency** command to set the interval to 0 milliseconds for an NQA operation, NQA does not generate any statistics group for the operation.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA operation.

   nqa entry *admin-name operation-tag*

3. Set the statistics collection interval.

   statistics interval *interval*

   The default setting is 60 minutes.

4. Set the maximum number of statistics groups that can be saved.

   statistics max-group *number*

   By default, the NQA client can save a maximum of two statistics groups for an operation.

   To disable the NQA statistics collection feature, set the *number* argument to 0.

5. Set the hold time of statistics groups.

   statistics hold-time *hold-time*

   The default setting is 120 minutes.

Configuring the saving of NQA history records

About this task

This task enables the NQA client to save NQA history records. You can use the **display nqa history** command to display the NQA history records.

Restrictions and guidelines

The NQA history record saving feature is not available for the following types of operations:

- ICMP jitter operation.
- UDP jitter operation.

- Voice operation.
- Path jitter operation.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA operation.

   nqa entry *admin-name operation-tag*

3. Enable the saving of history records for the NQA operation.

   history-record enable

   By default, this feature is enabled only for the UDP tracert operation.

4. Set the lifetime of history records.

   history-record keep-time *keep-time*

   The default setting is 120 minutes.

   A record is deleted when its lifetime is reached.

5. Set the maximum number of history records that can be saved.

   history-record number *number*

   The default setting is 50.

   When the maximum number of history records is reached, the system will delete the oldest record to save a new one.

Scheduling the NQA operation on the NQA client

About this task

The NQA operation runs between the specified start time and end time (the start time plus operation duration). If the specified start time is ahead of the system time, the operation starts immediately. If both the specified start and end time are ahead of the system time, the operation does not start. To display the current system time, use the **display clock** command.

Restrictions and guidelines

You cannot enter the operation type view or the operation view of a scheduled NQA operation.

A system time adjustment does not affect started or completed NQA operations. It affects only the NQA operations that have not started.

Procedure

1. Enter system view.

   system-view

2. Specify the scheduling parameters for an NQA operation.

   nqa schedule *admin-name operation-tag* **start-time** { *hh:mm:ss* [*yyyy/mm/dd* | *mm/dd/yyyy*] | **now** } **lifetime** { *lifetime* | **forever** } [**recurring**]

Configuring NQA templates on the NQA client

Restrictions and guidelines

Some operation parameters for an NQA template can be specified by the template configuration or the feature (such as load balancing) that uses the template. When both are specified, the parameters in the template configuration take effect.

NQA template tasks at a glance

To configure NQA templates, perform the following tasks:

1. Perform at least one of the following tasks:
   - Configuring the ARP template
   - Configuring the ICMP template
   - Configuring the IMAP template
   - Configuring the DNS template
   - Configuring the POP3 template
   - Configuring the SMTP template
   - Configuring the TCP template
   - Configuring the TCP half open template
   - Configuring the UDP template
   - Configuring the HTTP template
   - Configuring the HTTPS template
   - Configuring the FTP template
   - Configuring the RADIUS authentication template
   - Configuring the RTSP template
   - Configuring the SIP template
   - Configuring the SNMP template
   - Configuring the SNMP DCA template
   - Configuring the SSL template
   - Configuring the WAP template
2. (Optional.) Configuring optional parameters for the NQA template

Configuring the ARP template

About this task

A feature that uses the ARP template performs the ARP operation to test whether the ARP service is available on the destination device.

In the ARP operation, the NQA client sends an ARP request to the destination device. If the client receives an ARP reply, it determines that the ARP service is available on the destination device.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Create an ARP template and enter its view.

nqa template arp *name*

3. (Optional.) Specify the destination IP address for the probe packets.

   destination ip *ip-address*

   By default, no destination IP address is specified.

4. (Optional.) Specify the source IP address for the probe packets.

   source ip *ip-address*

   By default, the source IP address of the probe packets is the primary IP address of their output interface.

   The source IP address must be the IP address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the ICMP template

About this task

A feature that uses the ICMP template performs the ICMP operation to measure the reachability of a destination device. The ICMP template is supported on both IPv4 and IPv6 networks.

Procedure

1. Enter system view.

   system-view

2. Create an ICMP template and enter its view.

   nqa template icmp *name*

3. Specify the destination IP address for the operation.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination IP address is configured.

4. Specify the source IP address for ICMP echo requests. Choose one option as needed:

   - Use the IP address of the specified interface as the source IP address.

     source interface *interface-type interface-number*

     By default, the primary IP address of the output interface is used as the source IP address of ICMP echo requests.

     The specified source interface must be up.

   - Specify the source IPv4 address.

     source ip *ip-address*

     By default, the primary IPv4 address of the output interface is used as the source IPv4 address of ICMP echo requests.

     The specified source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   - Specify the source IPv6 address.

     source ipv6 *ipv6-address*

     By default, the IPv6 address of the output interface is used as the source IPv6 address of ICMP echo requests.

     The specified source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

5. Specify the next hop IP address for ICMP echo requests.

   IPv4:

   next-hop ip *ip-address*

   IPv6:

   next-hop ipv6 *ipv6-address*

   By default, no IP address of the next hop is configured.

6. Configure the probe result sending on a per-probe basis.

   reaction trigger per-probe

   By default, the probe result is sent to the feature that uses the template after three consecutive failed or successful probes.

   If you execute the **reaction trigger per-probe** and **reaction trigger probe-pass** commands multiple times, the most recent configuration takes effect.

   If you execute the **reaction trigger per-probe** and **reaction trigger probe-fail** commands multiple times, the most recent configuration takes effect.

7. (Optional.) Set the payload size for each ICMP request.

   data-size *size*

   The default setting is 100 bytes.

8. (Optional.) Specify the payload fill string for ICMP echo requests.

   data-fill *string*

   The default payload fill string is the hexadecimal string 00010203040506070809.

Configuring the IMAP template

About this task

A feature that uses the IMAP template performs the IMAP operation to determine the availability of the IMAP service on the IMAP server.

Before you perform an IMAP operation, enable the IMAP Server service on the IMAP server and configure related settings, including the login username, password, and mailbox name.

Procedure

1. Enter system view.

   system-view

2. Create an IMAP template and enter IMAP template view.

   nqa template imap name

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 143.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

source ip *ip-address*

By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

IPv6:

source ipv6 *ipv6-address*

By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. (Optional.) Specify the login username.

   username *username*

   By default, no login username is specified.

7. (Optional.) Specify the login password.

   password { **cipher** | **simple** } *string*

   By default, no login password is specified.

8. (Optional.) Specify the mailbox name.

   mailbox *mailbox-name*

   By default, mailbox **INBOX** is used.

Configuring the DNS template

About this task

A feature that uses the DNS template performs the DNS operation to determine the status of the server. The DNS template is supported on both IPv4 and IPv6 networks.

In DNS template view, you can specify the address expected to be returned. If the returned IP addresses include the expected address, the DNS server is valid and the operation succeeds. Otherwise, the operation fails.

Prerequisites

Create a mapping between the domain name and an address before you perform the DNS operation. For information about configuring the DNS server, see documents about the DNS server configuration.

Procedure

1. Enter system view.

   system-view

2. Create a DNS template and enter DNS template view.

   nqa template dns *name*

3. Specify the destination IP address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. Specify the destination port number for the probe packets.

destination port *port-number*

By default, the destination port number is 53.

5. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the source IPv4 address of the probe packets is the primary IPv4 address of their output interface.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. Specify the source port number for the probe packets.

   source port *port-number*

   By default, the NQA client randomly picks an unused port as the source port when the operation starts.

   For the operation to succeed, make sure the specified port number is not used by any services on the device. As a best practice, use the default value.

7. Specify the domain name to be translated.

   resolve-target *domain-name*

   By default, no domain name is specified.

8. Specify the domain name resolution type.

   resolve-type { **A** | **AAAA** }

   By default, the type is type A.

   A type A query resolves a domain name to a mapped IPv4 address, and a type AAAA query to a mapped IPv6 address.

9. (Optional.) Specify the IP address that is expected to be returned.

   IPv4:

   expect ip *ip-address*

   IPv6:

   expect ipv6 *ipv6-address*

   By default, no expected IP address is specified.

10. Configure the probe result sending on a per-probe basis.

    reaction trigger per-probe

    By default, the probe result is sent to the feature that uses the template after three consecutive failed or successful probes.

    If you execute the **reaction trigger per-probe** and **reaction trigger probe-pass** commands multiple times, the most recent configuration takes effect.

    If you execute the **reaction trigger per-probe** and **reaction trigger probe-fail** commands multiple times, the most recent configuration takes effect.

Configuring the POP3 template

About this task

A feature that uses the POP3 template performs the POP3 operation to determine the availability of the POP3 service on the POP3 server.

Before you perform a POP3 operation, enable the POP3 Server service on the POP3 server and configure related settings, including the login username and password.

Procedure

1. Enter system view.

   system-view

2. Create a POP3 template and enter POP3 template view.

   nqa template pop3 *name*

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 110.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. Specify the login username.

   username *username*

   By default, no login username is specified.

7. Specify the login password.

   password { **cipher** | **simple** } *string*

   By default, no login password is specified.

Configuring the SMTP template

About the SMTP template

A feature that uses the SMTP template performs the SMTP operation to determine the availability of the SMTP service on the SMTP server.

Procedure

1. Enter system view.

   system-view

2. Create a SMTP template and enter SMTP template view.

   nqa template smtp *name*

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 25.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the TCP template

About this task

A feature that uses the TCP template performs the TCP operation to test whether the NQA client can establish a TCP connection to a specific port on the server.

In TCP template view, you can specify the expected string to be returned. If you do not specify the expected string, the TCP operation tests only whether the client can establish a TCP connection to the server.

The TCP operation requires both the NQA server and the NQA client. Before you perform a TCP operation, configure a TCP listening service on the NQA server. For more information about the TCP listening service configuration, see "Configuring the NQA server."

Procedure

1. Enter system view.

   system-view

2. Create a TCP template and enter its view.

   nqa template tcp *name*

3. Specify the destination IP address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination IP address is specified.

   The destination address must be the same as the IP address of the TCP listening service configured on the NQA server.

4. Specify the destination port number for the operation.

   destination port *port-number*

   By default, no destination port number is specified.

   The destination port number must be the same as the port number of the TCP listening service on the NQA server.

5. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. (Optional.) Specify the payload fill string for the probe packets. Choose one option as needed:
   - Specify the payload fill string.

     data-fill *string*
   - Specify the hexadecimal payload fill string.

     hex-data-fill *hex* [**raw**]

   The default payload fill string is the hexadecimal string 00010203040506070809.

7. (Optional.) Configure the expected response string.
   - Configure the expected response string to determine a successful NQA operation:

     expect { **data** | **hex-data** } *string* [{ **offset** | **strict-offset** } *number*]
   - Configure the expected response string to determine a failed NQA operation:

     expect { **failed-data** | **hex-failed-data** } *string* [{ **offset** | **strict-offset** } *number*]

   By default, no expected response string is configured.

The expected response string check takes effect only if the **data-fill** or **hex-data-fill** command is also configured.

If you configure both commands, the **expect** { **failed-data** | **hex-failed-data** } command takes effect.

8. (Optional.) Enable the NQA client to send resource release notifications to the NQA server when the operation is complete.

   resource-release { **data-fill** | **hex-data-fill** } *string*

   By default, the NQA client does not send resource release notifications to the NQA server when an NQA operation is complete.

9. Set a TCP connection termination mode.

   disconnect-mode { **fin** | **rst** }

   By default, the TCP operation uses the RST mode to terminate TCP connections.

Configuring the TCP half open template

About this task

A feature that uses the TCP half open template performs the TCP half open operation to test whether the TCP service is available on the server. The TCP half open operation is used when the feature cannot get a response from the TCP server through an existing TCP connection.

The TCP half open operation works as follows:

- If port detection is disabled, the NQA client sends a TCP ACK packet to the server. If the client receives an RST packet, it considers that the TCP service is available on the server.
- If port detection is enabled, the NQA client sends a TCP SYN packet to the server. If the client receives a SYN-ACK packet, it considers that the destination TCP port is operating correctly on the server.

Procedure

1. Enter system view.

   system-view

2. Create a TCP half open template and enter its view.

   nqa template tcphalfopen *name*

3. (Optional.) Enable port detection.

   port-detect enable

   By default, port detection is disabled.

   To use the port detection feature, you must specify a destination port number for the TCP half open template.

4. Specify the destination IP address of the operation.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination IP address is specified.

   The destination address must be the same as the IP address of the listening service configured on the NQA server.

5. Specify the destination port number.

   destination port *port-number*

   By default, no destination port number is specified.

The destination port number must be the same as the port number of the listening service on the NQA server.

For the port detection feature to take effect, the destination port number must be specified.

6. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

7. Specify the output interface for the probe packets.

   out interface *interface-type interface-number*

   By default, the NQA client determines the output interface based on the routing table lookup.

8. Specify the next hop IP address for the probe packets.

   IPv4:

   next-hop ip *ip-address*

   IPv6:

   next-hop ipv6 *ipv6-address*

   By default, the IP address of the next hop is configured.

9. Configure the probe result sending on a per-probe basis.

   reaction trigger per-probe

   By default, the probe result is sent to the feature that uses the template after three consecutive failed or successful probes.

   If you execute the **reaction trigger per-probe** and **reaction trigger probe-pass** commands multiple times, the most recent configuration takes effect.

   If you execute the **reaction trigger per-probe** and **reaction trigger probe-fail** commands multiple times, the most recent configuration takes effect.

Configuring the UDP template

About this task

A feature that uses the UDP template performs the UDP operation to test the following items:

- Reachability of a specific port on the NQA server.
- Availability of the requested service on the NQA server.

The UDP operation requires both the NQA server and the NQA client. Before you perform a UDP operation, configure a UDP listening service on the NQA server. For more information about the UDP listening service configuration, see "Configuring the NQA server."

(For devices that do not support UDP port detection.) A UDP operation result varies by the configuration of the expected string:

- If the expected string that determines a failed UDP operation is configured, the UDP operation fails when either of the conditions is met:
  - The UDP response string contains the expected string.
  - The UDP response string is shorter than the expected string.

  If none conditions are met, the operation is successful.
- If the expected string that determines a successful UDP operation is configured, the UDP operation is successful when the UDP response packet matches the expected string. Otherwise, the operation fails.
- If no expected string is configured, the UDP operation is successful when the client receives the response packet from the server.

(For devices that support UDP port detection.) The UDP port detection tests the availability of the requested UDP service on the peer port. A UDP operation result varies by the configuration of the expected string and the status of the port detection:

- With port detection enabled for the UDP operation, the operation succeeds if the NQA client does not receive any ICMP port unreachable messages within the probe timeout time. If the client receives an ICMP port unreachable message, the UDP operation fails.
- With port detection disabled for the UDP operation, the following rules apply:
- If the expected string that determines a failed UDP operation is configured, the UDP operation fails when either of the conditions is met:
  - The UDP response string contains the expected string.
  - The UDP response string is shorter than the expected string.

  If none conditions are met, the operation is successful.
  - If the expected string that determines a successful UDP operation is configured, the UDP operation is successful when the UDP response packet matches the configured expected string. Otherwise, the operation fails.
  - If no expected string is configured, the UDP operation is successful when the client receives the response packet from the server.

Restrictions and guidelines

If the destination device is an NSFOCUS device, you must also perform the following tasks for the UDP operation:

- Execute the **`ip unreachables enable`** command on the destination device to enable sending ICMP destination unreachable messages. For more information about the **`ip unreachables enable`** command, see IP performance optimization commands in *Layer 3 IP Services Command Reference*.
- Execute the **`data-fill`** or **`hex-data-fill`** command on the NQA client with the **`raw`** keyword specified. The specified payload fill string can be any value in the value range.

Procedure

1. Enter system view.

   `system-view`
2. Create a UDP template and enter its view.

   `nqa template udp` *`name`*
3. (Optional.) Enable port detection for the UDP operation.

   `port-detect enable`

   By default, port detection is disabled.
4. Specify the destination IP address of the operation.

   IPv4:

   `destination ip` *`ip-address`*

IPv6:

destination ipv6 *ipv6-address*

By default, no destination IP address is specified.

The destination address must be the same as the IP address of the listening service configured on the NQA server.

5. Specify the destination port number for the probe packets.

   destination port *port-number*

   By default, no destination port number is specified.

   The destination port number must be the same as the port number of the listening service on the NQA server.

6. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

7. Specify the payload fill string for the probe packets. Choose one option as needed:

   - Specify the payload fill string.

     data-fill *string* [**raw**]

     The default payload fill string is the hexadecimal string 00010203040506070809.

   - Specify the hexadecimal payload fill string.

     hex-data-fill *hex* [**raw**]

   The default payload fill string is the hexadecimal string 00010203040506070809.

   For port detection to take effect for the UDP operation, specify the **raw** keyword in the **data-fill** or **hex-data-fill** command, and the string is not required to be format compliant.

8. (Optional.) Set the payload size for the probe packets.

   data-size *size*

   The default setting is 100 bytes.

9. (Optional.) Configure the expected response string.

   - Configure the expected response string to determine a successful NQA operation:

     expect { **data** | **hex-data** } *string* [{ **offset** | **strict-offset** } *number*]

   - Configure the expected response string to determine a failed NQA operation:

     expect { **failed-data** | **hex-failed-data** } *string* [{ **offset** | **strict-offset** } *number*]

   By default, no expected response string is configured.

   The expected response string check takes effect only if the **data-fill** or **hex-data-fill** command is also configured.

If you configure both commands, the **expect { failed-data | hex-failed-data }** command takes effect.

10. (Optional.) Enable the NQA client to send resource release notifications to the NQA server when the operation is complete.

    resource-release { data-fill | hex-data-fill } *string*

    By default, the NQA client does not send resource release notifications to the NQA server when an NQA operation is complete.

Configuring the HTTP template

About this task

A feature that uses the HTTP template performs the HTTP operation to measure the time it takes the NQA client to obtain data from an HTTP server.

The expected string is checked only when the data is configured and the HTTP response contains the Content-Length field in the HTTP header.

The status code of the HTTP packet is a three-digit field in decimal notation, and it includes the status information for the HTTP server. The first digit defines the class of response.

Prerequisites

Before you perform the HTTP operation, you must configure the HTTP server.

Procedure

1. Enter system view.

   system-view

2. Create an HTTP template and enter its view.

   nqa template http *name*

3. Specify the URL of the HTTP proxy server.

   proxy-url *url*

   By default, the URL of the HTTP proxy server is not specified.

   The URL of the HTTP proxy server is in the format of http://*host* or http://*host:port*.

   This command is required if an HTTP proxy server is required for Internet access.

   After the HTTP proxy server URL is specified, the NQA client will send probe packets to the HTTP proxy server, which acts on behalf of the HTTP server.

4. Specify the URL of the destination HTTP server for the HTTP template.

   url *url*

   By default, the URL of the destination HTTP server is not specified for an HTTP template.

   The URL can be in one of the following formats:

   - http://host/resource
   - http://host:port/resource

5. Specify an HTTP login username.

   username *username*

   By default, no HTTP login username is specified.

6. Specify an HTTP login password.

   password { cipher | simple } *string*

   By default, no HTTP login password is specified.

7. Specify the HTTP version.

   version { v1.0 | v1.1 }

By default, HTTP 1.0 is used.

8. Specify the HTTP operation type.

   operation { **get** | **post** | **raw** }

   By default, the HTTP operation type is **get**.

   If you set the operation type to raw, the client pads the content configured in raw request view to the HTTP request to send to the HTTP server.

9. Configure the content of the HTTP raw request.

   a. Enter raw request view.

      raw-request

      Every time you enter raw request view, the previously configured raw request content is cleared.

   b. Enter or paste the request content.

      By default, no request content is configured.

      For successful HTTP operations, make sure the raw request content is valid and does not contain command aliases configured by using the **alias** command. For more information about the **alias** command, see CLI commands in *Fundamentals Command Reference*.

   c. Return to HTTP template view.

      quit

      The system automatically saves the configuration in raw request view before it returns to HTTP template view.

   This step is required only when the operation type is set to **raw**.

10. Specify the source IP address for the probe packets.

    IPv4:

    source ip *ip-address*

    By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

    The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

    IPv6:

    source ipv6 *ipv6-address*

    By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

    The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

11. (Optional.) Configure the expected status codes to determine a successful NQA operation.

    expect status *status-list*

    By default, no expected status code is configured to determine a successful NQA operation.

    Do not configure both this command and the **expect failed-status** command.

12. (Optional.) Configure the expected status codes to determine a failed NQA operation.

    expect failed-status *status-list*

    By default, no expected status code is configured to determine a failed NQA operation.

    Do not configure both this command and the **expect status** command.

13. (Optional.) Configure the expected response string to determine a successful NQA operation.

    expect data *string* [{ **offset** | **strict-offset** } *number*]

    By default, no expected response string is configured to determine a successful NQA operation.

Do not configure both this command and the **expect failed-data** command.

14. (Optional.) Configure the expected response string to determine a failed NQA operation.

    expect failed-data *string* [{ **offset** | **strict-offset** } *number*]

    By default, no expected response string is configured to determine a failed NQA operation..

    Do not configure both this command and the **expect data** command.

Configuring the HTTPS template

About this task

A feature that uses the HTTPS template performs the HTTPS operation to measure the time it takes for the NQA client to obtain data from an HTTPS server.

The expected string is checked only when the expected string is configured and the HTTPS response contains the Content-Length field in the HTTPS header.

The status code of the HTTPS packet is a three-digit field in decimal notation, and it includes the status information for the HTTPS server. The first digit defines the class of response.

Prerequisites

Before you perform the HTTPS operation, configure the HTTPS server and the SSL client policy for the SSL client. For information about configuring SSL client policies, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create an HTTPS template and enter its view.

   nqa template https *name*

3. Specify the URL of the HTTPS proxy server.

   proxy-url *url*

   By default, the URL of the HTTPS proxy server is not specified.

   The URL of the HTTPS proxy server is in the format of https://*host* or https://*host:port*.

   This command is required if an HTTPS proxy server is required for Internet access.

   After the HTTPS proxy server URL is specified, the NQA client will send probe packets to the HTTPS proxy server, which acts on behalf of the HTTPS server.

4. Specify the URL of the destination HTTPS server for the HTTPS template.

   url *url*

   By default, the URL of the destination HTTPS server is not specified for an HTTPS template.

   Enter the URL in one of the following formats:

   - https://host/resource
   - https://host:port/resource

5. Specify an HTTPS login username.

   username *username*

   By default, no HTTPS login username is specified.

6. Specify an HTTPS login password.

   password { **cipher** | **simple** } *string*

   By default, no HTTPS login password is specified.

7. Specify an SSL client policy.

ssl-client-policy *policy-name*

By default, no SSL client policy is specified.

8. Specify the HTTPS version.

version { **v1.0** | **v1.1** }

By default, HTTPS 1.0 is used.

9. Specify the HTTPS operation type.

operation { **get** | **post** | **raw** }

By default, the HTTPS operation type is **get**.

If you set the operation type to raw, the client pads the content configured in raw request view to the HTTPS request to send to the HTTPS server.

10. Configure the content of the HTTPS raw request.

a. Enter raw request view.

raw-request

Every time you enter raw request view, the previously configured raw request content is cleared.

b. Enter or paste the request content.

By default, no request content is configured.

For successful HTTP operations, make sure the raw request content is valid and does not contain command aliases configured by using the **alias** command. For more information about the **alias** command, see CLI commands in *Fundamentals Command Reference*.

c. Return to HTTPS template view.

quit

The system automatically saves the configuration in raw request view before it returns to HTTPS template view.

This step is required only when the operation type is set to **raw**.

11. Specify the source IP address for the probe packets.

IPv4:

source ip *ip-address*

By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

IPv6:

source ipv6 *ipv6-address*

By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

12. (Optional.) Configure the expected status codes to determine a successful NQA operation.

expect status *status-list*

By default, no expected status code is configured to determine a successful NQA operation.

Do not configure both this command and the **expect failed-status** command.

13. (Optional.) Configure the expected status codes to determine a failed NQA operation.

expect failed-status *status-list*

By default, no expected status code is configured to determine a failed NQA operation.

Do not configure both this command and the **expect status** command.

14. (Optional.) Configure the expected response string to determine a successful NQA operation.

    expect data *string* [{ **offset** | **strict-offset** } *number*]

    By default, no expected response string is configured to determine a successful NQA operation.

    Do not configure both this command and the **expect failed-data** command.

15. (Optional.) Configure the expected response string to determine a failed NQA operation.

    expect failed-data *string* [{ **offset** | **strict-offset** } *number*]

    By default, no expected response string is configured to determine a failed NQA operation..

    Do not configure both this command and the **expect data** command.

Configuring the FTP template

About this task

A feature that uses the FTP template performs the FTP operation. The operation measures the time it takes the NQA client to transfer a file to or download a file from an FTP server.

Configure the username and password for the FTP client to log in to the FTP server before you perform an FTP operation. For information about configuring the FTP server, see *Fundamentals Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create an FTP template and enter its view.

   nqa template ftp *name*

3. Specify an FTP login username.

   username *username*

   By default, no FTP login username is specified.

4. Specify an FTP login password.

   password { **cipher** | **simple** } *sting*

   By default, no FTP login password is specified.

5. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. Set the data transmission mode.

   mode { **active** | **passive** }

   The default mode is **active**.

7. Specify the FTP operation type.

   operation { **get** | **put** }

   By default, the FTP operation type is **get**, which means obtaining files from the FTP server.

8. Specify the destination URL for the FTP template.

   url *url*

   By default, no destination URL is specified for an FTP template.

   Enter the URL in one of the following formats:

   - ftp://*host*/*filename*.
   - ftp://*host*:*port*/*filename*.

   When you perform the **get** operation, the file name is required.

   When you perform the **put** operation, the *filename* argument does not take effect, even if it is specified. The file name for the **put** operation is determined by using the **filename** command.

9. Specify the name of a file to be transferred.

   filename *filename*

   By default, no file is specified.

   This task is required only for the **put** operation.

   The configuration does not take effect for the **get** operation.

Configuring the RADIUS authentication template

About this task

A feature that uses the RADIUS authentication template performs the RADIUS authentication operation to check the availability of the authentication service on the RADIUS server.

The RADIUS authentication operation workflow is as follows:

1. The NQA client sends an authentication request (Access-Request) to the RADIUS server. The request includes the username and the password. The password is encrypted by using the MD5 algorithm and the shared key.
2. The RADIUS server authenticates the username and password.
   - If the authentication succeeds, the server sends an Access-Accept packet to the NQA client.
   - If the authentication fails, the server sends an Access-Reject packet to the NQA client.
3. The NQA client determines the availability of the authentication service on the RADIUS server based on the response packet it received:
   - If an Access-Accept packet is received, the authentication service is available on the RADIUS server.
   - If an Access-Reject packet is received, the authentication service is not available on the RADIUS server.

Prerequisites

Before you configure the RADIUS authentication template, specify a username, password, and shared key on the RADIUS server. For more information about configuring the RADIUS server, see AAA in *Security Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a RADIUS authentication template and enter its view.

nqa template radius *name*

3. Specify the destination IP address of the operation.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination IP address is specified.

4. Specify the destination port number for the operation.

   destination port *port-number*

   By default, the destination port number is 1812.

5. Specify a username.

   username *username*

   By default, no username is specified.

6. Specify a password.

   password { **cipher** | **simple** } *string*

   By default, no password is specified.

7. Specify a shared key for secure RADIUS authentication.

   key { **cipher** | **simple** } *string*

   By default, no shared key is specified for RADIUS authentication.

8. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the RADIUS accounting template

About this task

A feature that uses the RADIUS accounting template performs the RADIUS accounting operation to check the availability of the accounting service on the RADIUS server.

The RADIUS accounting operation workflow is as follows:

1. The NQA client sends a stop-accounting request (Accounting-Request) to the RADIUS server.
2. The RADIUS server returns an acknowledgment (Accounting-Response) to the NQA client.
3. The NQA client determines the availability of the accounting service on the RADIUS server based on whether it can receive the accounting response packet from the RADIUS server.
   - If the accounting response packet is received, the accounting service is available on the RADIUS server.

- If no accounting response packet is received, the accounting service is not available on the RADIUS server.

Prerequisites

Before you configure the RADIUS accounting template, specify a username and the shared key on the RADIUS server. For more information about configuring the RADIUS server, see AAA in *Security Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a RADIUS accounting template and enter its view.

   nqa template radius-account *name*

3. Specify the destination IP address of the operation.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination IP address is configured.

4. Specify the destination port number for the operation.

   destination port *port-number*

   By default, the destination port number is 1813.

5. Specify a username.

   username *username*

   By default, no username is specified.

6. Specify a shared key for secure RADIUS accounting.

   key { **cipher** | **simple** } *string*

   By default, no shared key is specified for RADIUS accounting.

7. Specify the source IP address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the RTSP template

About this task

The Real Time Streaming Protocol (RTSP) is a network control protocol designed for real-time control (such as play/pause and forward/playback) of media streaming on the Internet.

A feature that uses the RTSP template performs the RTSP operation to test the availability of the RTSP server.

Procedure

1. Enter system view.

   system-view

2. Create an RTSP template and enter RTSP template view.

   nqa template rtsp *name*

3. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

4. (Optional.) Specify the request method.

   - Specify the OPTIONS method.

     request-method options

   - Specify the DESCRIBE method.

     request-method describe

   By default, the OPTIONS method is used.

5. (Optional.) Configure the expected status codes to determine a successful NQA operation.

   expect status *status-list*

   By default, no expected status code is configured to determine a successful NQA operation.

6. (Optional.) Specify the destination URL.

   url *url*

   By default, no destination URL is specified.

   Valid formats for the destination URL are:

   - rtsp://*host*/*resource*
   - rtsp: //*host*:*port*/*resource*

Configuring the SIP template

About this task

The Session Initiation Protocol (SIP) is a communications protocol for signaling that controls multimedia communication sessions over IP networks. SIP messages can be transported over UDP or TCP.

A feature that uses the SIP template performs the SIP operation to test the availability of the SIP service on the SIP server.

Procedure

1. Enter system view.

   system-view

2. Create a SIP template and enter SIP template view.

   nqa template sip *name*

3. Specify the transport protocol.

   transport-protocol { **tcp** | **udp** }

   By default, the UDP transport protocol is used.

4. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

5. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 5060.

6. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

7. (Optional.) Configure the expected status codes to determine a successful NQA operation.

   expect status *status-list*

   By default, no expected status code is configured to determine a successful NQA operation.

Configuring the SNMP template

About this task

A feature that uses the SNMP template performs the SNMP operation to determine the availability of the SNMP service on an SNMP agent.

Procedure

1. Enter system view.

   system-view

2. Create an SNMP template and enter SNMP template view.

   nqa template snmp *name*

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 161.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. (Optional.) Specify the community name for SNMPv1 or SNMPv2c probe packets.

   community read { **cipher** | **simple** } *community-name*

   By default, the SNMP template uses community name **public**.

   Make sure the specified community name is the same as the community name configured on the SNMP agent.

Configuring the SNMP DCA template

About this task

A feature that uses the SNMP Data Collection Agent (DCA) template performs the SNMP DCA operation to measure the performance of a device running an SNMP agent. The SNMP DCA operation involves the following steps:

1. The NQA client sends queries to collect SNMP object values from the SNMP agent.

   The query message contains the OID that identifies the object to collect on the SNMP agent. Typically, the CPU, memory, and disk usage objects are collected.

2. The NQA client determines the performance of the device based on the collected object values and the threshold and weight values configured for the objects. Then, the NQA client reports the information to the associated feature for the feature to act accordingly.

Before you start the SNMP DCA operation, you must configure the SNMP agent.

Procedure

1. Enter system view.

   system-view

2. Create an SNMP DCA template and enter its view.

   nqa template snmpdca *name*

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 161.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. (Optional.) Specify the SNMP community name.

   community read { **cipher** | **simple** } *community-name*

   The default SNMP community name is **public**.

   Make sure the specified community name is the same as the community name configured on the SNMP agent.

7. (Optional.) Specify the SNMP agent type.

   agent-type { **net-snmp** | **user-defined** | **windows** }

   The default SNMP agent type is Net-SNMP.

8. (Optional.) Specify the SNMP version.

   version { **v1** | **v2c** }

   By default, SNMPv1 is used.

9. (Optional.) Specify the threshold and weight for the CPU usage object.

   cpu { **threshold** *threshold-value* | **weight** *weight-value* }

   By default, the CPU usage threshold is 80 and the weight is 3.

10. (Optional.) Specify the threshold and weight for the memory usage object.

    memory { **threshold** *threshold-value* | **weight** *weight-value* } *

    By default, the memory usage threshold is 70 and the weight is 2.

11. (Optional.) Specify the threshold and weight for the disk usage object.

    disk { **threshold** *threshold-value* | **weight** *weight-value* } *

    By default, the disk usage threshold is 90 and the weight is 4.

12. (Optional.) Configure a custom SNMP object for the SNMP DCA to collect and set the threshold and weight for the object.

    oid *oid* **threshold** *threshold-value* **weight** *weight-value*

By default, an SNMP DCA template does not contain custom SNMP objects.

This step is required if the **`user-defined`** SNMP agent type is used.

Configuring the SSL template

About this task

A feature that uses the SSL template performs the SSL operation to measure the time required to establish an SSL connection to an SSL server.

Prerequisites

Before you configure the SSL template, you must configure the SSL client policy. For information about configuring SSL client policies, see *Security Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Create an SSL template and enter its view.

   `nqa template ssl` *`name`*

3. Specify the destination IP address of the operation.

   IPv4:

   `destination ip` *`ip-address`*

   IPv6:

   `destination ipv6` *`ipv6-address`*

   By default, no destination IP address is specified.

4. Specify the destination port number for the operation.

   `destination port` *`port-number`*

   By default, the destination port number is not specified.

5. Specify an SSL client policy.

   `ssl-client-policy` *`policy-name`*

   By default, no SSL client policy is specified.

6. Specify the source IP address for the probe packets.

   IPv4:

   `source ip` *`ip-address`*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IP address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   `source ipv6` *`ipv6-address`*

   By default, the IPv6 address of the output interface is used as the source IPv6 address of the probe packets.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

Configuring the WAP template

About this task

Wireless Application Protocol (WAP) is a technical standard for accessing information over a mobile wireless network. It describes a protocol suite allowing the interoperability of WAP equipment, and software with different network technologies.

A feature that uses the WAP template performs the WAP operation to determine the availability of the WAP gateway.

Before you start the WAP operation, you must configure the WAP gateway.

Procedure

1. Enter system view.

   system-view

2. Create an WAP template and enter its view.

   nqa template wap *name*

3. (Optional.) Specify the destination address for the probe packets.

   IPv4:

   destination ip *ip-address*

   IPv6:

   destination ipv6 *ipv6-address*

   By default, no destination address is specified.

4. (Optional.) Specify the destination port number for the probe packets.

   destination port *port-number*

   The default destination port number is 161.

5. (Optional.) Specify the source address for the probe packets.

   IPv4:

   source ip *ip-address*

   By default, the primary IPv4 address of the output interface is used as the source IPv4 address of the probe packets.

   The source IPv4 address must be the IPv4 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

   IPv6:

   source ipv6 *ipv6-address*

   By default, the source IPv6 address of the probe packets is the IPv6 address of their output interface.

   The source IPv6 address must be the IPv6 address of a local interface, and the interface must be up. Otherwise, no probe packets can be sent out.

6. (Optional.) Specify the destination URL.

   url *url*

   By default, no destination URL is specified.

   Valid formats for the destination URL are:

   - http://host/resource
   - http://host:port/resource
   - https://host/resource
   - https://host:port/resource

7. (Optional.) Configure the expected response string to determine a successful NQA operation.

expect { **data** | **hex-data** } *string* [{ **offset** | **strict-offset** } *number*]

By default, no expected response string is configured to determine a successful NQA operation.

This step is required if the **url** command is configured.

Configuring optional parameters for the NQA template

Restrictions and guidelines

Unless otherwise specified, the following optional parameters apply to all types of NQA templates.

The parameter settings take effect only on the current NQA template.

Procedure

1. Enter system view.

   system-view

2. Enter the view of an existing NQA template.

   nqa template { **arp** | **dns** | **ftp** | **http** | **https** | **icmp** | **imap** | **pop3** | **radius** | **radius-account** | **rtsp** | **sip** | **smtp** | **snmp** | **snmpdca** | **ssl** | **tcp** | **tcphalfopen** | **udp** | **wap** } *name*

3. Configure a description.

   description *text*

   By default, no description is configured.

4. Set the interval at which the NQA operation repeats.

   frequency *interval*

   The default setting is 5000 milliseconds.

   When the interval expires, but the operation is not completed or is not timed out, the next operation does not start.

5. Specify the adjusted interval for NQA to start two consecutive NQA operations after a failed operation.

   frequency-adjustment *adj-interval*

   By default, no adjusted interval is specified.

6. Set the probe timeout time.

   probe timeout *timeout*

   The default setting is 3000 milliseconds.

7. Set the TTL for the probe packets.

   ttl *value*

   The default setting is 20.

   This command is not available for the ARP template.

8. Set the ToS value in the IP header of the probe packets.

   tos *value*

   The default setting is 0.

   This command is not available for the ARP template.

9. Specify the VPN instance where the operation is performed.

   vpn-instance *vpn-instance-name*

   By default, the operation is performed on the public network.

10. Set the number of consecutive successful probes to determine a successful operation event.

reaction trigger probe-pass *count*

The default setting is 3.

If the number of consecutive successful probes for an NQA operation is reached, the NQA client notifies the feature that uses the template of the successful operation event.

11. Set the number of consecutive probe failures to determine an operation failure.

reaction trigger probe-fail *count*

The default setting is 3.

If the number of consecutive probe failures for an NQA operation is reached, the NQA client notifies the feature that uses the NQA template of the operation failure.

Display and maintenance commands for NQA

Execute **display** commands in any view.

Task	Command
Display history records of NQA operations.	**display nqa history** [*admin-name operation-tag*]
Display the current monitoring results of reaction entries.	**display nqa reaction counters** [*admin-name operation-tag* [*item-number*]]
Display the most recent result of the NQA operation.	**display nqa result** [*admin-name operation-tag*]
Display NQA server status.	**display nqa server status**
Display NQA statistics.	**display nqa statistics** [*admin-name operation-tag*]

Configuring TWAMP Light

About TWAMP Light

Two-Way Active Measurement Protocol (TWAMP) defines a standard to measure the network performance between network devices on an IP network. It uses UDP packets to measure the two-way Frame Transfer Delay (FTD), Frame Delay Variation (FDV), and Frame Loss Ratio (FLR). The TWAMP Light provides a simple structure of TWAMP. It simplifies the control protocol for establishing performance measurement sessions and improves test performance.

TWAMP Light architecture

TWAMP Light uses the client-server model.

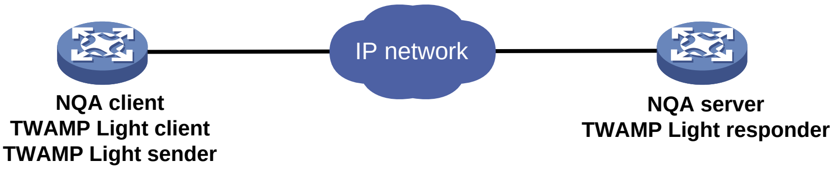
Figure 2 describes the TWAMP Light roles and the typical network diagram.

On the client, the following roles are configured:

- **TWAMP Light client**—Configures TWAMP Light test sessions.
- **TWAMP Light sender**—Starts and stops TWAMP Light test sessions, and collect statistics.

On the server, the TWAMP Light responder, also known as the destination device, is configured. The responder reflects the packets back to the TWAMP Light sender. You must enable the NQA server on the destination device, and create the TWAMP Light responder and test sessions.

Figure 2 Network diagram



TWAMP Light operating mechanism

A TWAMP Light test contains a set of parameters for a test session such as the source IP address and destination IP address.

Each TWAMP Light test session is uniquely identified by the test session ID. You can create and run multiple test sessions on one TWAMP Light client.

The TWAMP Light client and TWAMP Light responder interact as follows:

1. The TWAMP Light client constructs TWAMP Light test packets, and sends them to the TWAMP Light responder.
2. The TWAMP Light responder reflects the test packets to the TWAMP Light client.
3. Upon receiving the reflected packets, the TWAMP Light client calculates the packet loss ratio and round-trip time to determine the service quality from source to destination.

After a TWAMP Light test starts, the TWAMP light client runs the tests permanently or repeats the test at the specified interval. Each test sends one test packet. You can set the test duration and number of test packets to be sent.

Threshold monitoring

Threshold monitoring enables the TWAMP Light client to take a predefined action when the TWAMP Light test performance metrics violate the specified thresholds.

The TWAMP Light test monitors the following metrics:

- Two-way frame delay variation.
- Two-way frame transfer delay.
- Two-way frame loss rate.

In a TWAMP test, the device monitors the test result, and starts the monitoring time when either of the following conditions is met:

- The monitoring result goes beyond the threshold upper limit.
- The monitoring result drops below the threshold lower limit from a monitoring result higher than the lower limit.

If either condition is always true during the monitoring time, a threshold violation occurs and a trap or inform message is generated and sent to the NMS. You set the monitoring time by using the corresponding command.

Protocols and standards

- RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)*

Restrictions: Hardware compatibility with TWAMP Light

Models	**TWAMP Light compatibility**
NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB680, NFNX3-HDB1080	Yes
NFNX3-HDB1180, NFNX3-HDB1480	No

Restrictions and guidelines: TWAMP Light configuration

To avoid probe failures, follow these restrictions and guidelines when configuring the listening port on the TWAMP Light server and the TWAMP Light client:

- Do not specify a well-known port.
- Make sure the specified port number is not used by any services on the device.
  - To obtain the IPv4 addresses and the port numbers in use on this device, see the **Local Addr:port** field in the output from the **`display tcp`** and **`display udp`** commands.
  - To obtain the IPv6 addresses and the port numbers in use on this device, see the **LAddr->port** field in the output from the **`display ipv6 tcp`** and **`display ipv6 udp`** commands.

The destination port configured for the test (with the **destination port** command) on the TWAMP Light client must be the same as the listening port configured on the server.

TWAMP Light tasks at a glance

To configure TWAMP Light, perform the following tasks:

1. Configuring the TWAMP Light server
2. Configuring the TWAMP Light client
3. (Optional.) Configuring threshold monitoring
4. Start the test on the TWAMP Light sender
5. (Optional.) Stop the test on the TWAMP Light sender

Configuring the TWAMP Light server

1. Enter system view.
   system-view
2. Enable the TWAMP Light responder on the NQA server and enter its view.
   nqa twamp-light responder
3. Create a test session on the TWAMP Light responder.
   test-session *session-id* [**interface** *interface-type interface-number* [**service-instance** *instance-id*]] { { **ip** | **ipv6** } **destination** *address* **source** *address* **destination-port** *port-number* **source-port** *port-number* [**vpn-instance** *vpn-instance-name*] | **destination-mac** *mac-address* **source-mac** *mac-address* } * [**vlan** { *vlan-id* | **s-vid** *vlan-id* **c-vid** *vlan-id* } | **timestamp-format** { **ntp** | **ptp** } | **description** *text*] *
4. Return to system view.
   quit
5. Enable the NQA server.
   nqa server enable
   By default, the NQA server is disabled.

Configuring the TWAMP Light client

Restrictions and guidelines

In the TWAMP Light test, a test session is identified by the combination of its address and port number. To ensure the test result, you cannot specify the same combination of address and port number for multiple test sessions.

Procedure

1. Enter system view.
   system-view
2. Enable the NQA client.
   nqa agent enable
   By default, the NQA client is enabled.
   After the NQA client is enabled, the TWAMP Light test session configuration can takes effect.
3. Enable the TWAMP Light client and enter its view.

nqa twamp-light client

4. Create a test session on the TWAMP Light client and enter the client-session view.

   test-session *session-id*

5. (Optional.) Specify the description for the test.

   description *text*

   By default, no description is specified for the test.

6. Specify the IP address and port number for the TWAMP Light test session.

   a. Specify the source IP address for the probe packets.

      IPv4:

      source ip *ip-address*

      By default, no source IPv4 address for the probe packets is specified.

      IPv6:

      source ipv6 *ipv6-address*

      By default, no source IPv6 address for the probe packets is specified.

   b. Specify the destination IP address for the probe packets.

      IPv4:

      destination ip *ipv4-address*

      By default, no destination IPv4 address for the probe packets is specified.

      IPv6:

      destination ipv6 *ipv6-address*

      By default, no destination IPv6 address for the probe packets is specified.

   c. Specify the source interface for the probe packets.

      source interface *interface-type interface-number* [**service-instance** *instance-id*]

      By default, no source interface for the probe packets is specified.

      The specified source interface must be up.

   d. Specify the source port number for the probe packets.

      source port *port-number*

      By default, no source port number for the probe packets is specified.

      For TWAMP Light tests, you must configure this command. For the test to succeed, make sure the specified port number is not used by any services on the device.

   e. Specify the destination port number for the probe packets.

      destination port *port-number*

      By default, no destination port number for the probe packets is specified.

   f. Specify the source MAC address for the probe packets.

      source mac *mac-address*

      By default, no source MAC address for the probe packets is specified.

   g. Specify the destination MAC address for the probe packets.

      destination mac *mac-address*

      By default, no destination MAC address for the probe packets is specified.

   h. (Optional.) Specify the VPN instance where the test is performed.

      vpn-instance *vpn-instance-name*

      By default, no VPN instance is specified. The test is performed on the public network.

7. Specify the timestamp format for the TWAMP Light test session.

timestamp-format { **ntp** | **ptp** }

By default, the timestamp format for the TWAMP Light test is PTP.

8. Specify the payload parameters for the TWAMP Light probe packets.
   - Set the payload size for each probe packet.

     data-size *size*

     The default payload size is 142 bytes.
   - Specify the payload fill string for each probe packet.
     - **data-fill** *string*
     - **hex-data-fill** *hex*

     The default hexadecimal packet payload fill string is 00010203040506070809.

     The two commands have the same function. If you execute them multiple times, the most recent configuration takes effect.
9. (Optional.) Set the priority for the probe packets.
   - Set the 802.1p priority for the probe packets.

     priority 8021p *value*

     By default, the 802.1p priority of the probe packets is 0.
   - Set the ToS value in the IP header of the probe packets.

     tos *value*

     The default setting is 0.
10. (Optional.) Specify the ID of the VLAN to which the probe packets belong.

    vlan { *vlan-id* | **s-vid** *vlan-id* **c-vid** *vlan-id* }

    By default, no VLAN ID is specified.

Configuring threshold monitoring

1. Enter system view.

   system-view
2. Enable the TWAMP Light client and enter its view.

   nqa twamp-light client
3. Create a test session on the TWAMP Light client and enter the client-session view.

   test-session *session-id*
4. Configure a reaction entry. Choose the following tasks as needed:
   - Configure a reaction entry for monitoring the two-way delay.

     reaction *item-number* **checked-element two-way-delay threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     By default, no reaction entry is configured for monitoring the two-way delay.
   - Configure a reaction entry for monitoring the two-way packet loss.

     reaction *item-number* **checked-element two-way-loss threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

     By default, no reaction entry is configured for monitoring the two-way packet loss.
   - Configure a reaction entry for monitoring the two-way jitter.

reaction *item-number* **checked-element two-way-jitter threshold-value** *upper-threshold lower-threshold* [**action-type** { **none** | **trap-only** }]

By default, no reaction entry is configured for monitoring the two-way jitter.

Start the test on the TWAMP Light sender

Restrictions and guidelines

In the TWAMP Light test, a test session is identified by the combination of source IP address, source port number, destination IP address, and destination port number. To ensure the test result, do not specify the same combination for multiple test sessions.

With the **data-fill** command configured, the packet sending interval cannot be 10 or 100 milliseconds.

Procedure

1. Enter system view.

   system-view

2. Enable the TWAMP Light sender and enter its view.

   nqa twamp-light sender

3. Start a TWAMP Light test.

   start test-session *session-id* { **permanent** | **duration** *duration* | **packet-count** *count* } [**tx-interval** { **10** | **100** | **1000** | **30000** }] [**timeout** *timeout*] [[**statistics-interval** *statistics-interval*] **monitor-time** *time*]

Stop the test on the TWAMP Light sender

1. Enter system view.

   system-view

2. Enter the TWAMP Light sender view.

   nqa twamp-light sender

3. Stop the TWAMP Light test.

   stop { **all** | **test-session** *session-id* }

Display and maintenance commands for TWAMP Light

Execute **display** commands in any view on the TWAMP Light responder.

Task	Command
Display test sessions on the TWAMP Light responder.	**display nqa twamp-light responder** [**test-session** *session-id*]

Execute **display** commands in any view and **reset** commands in user view on the TWAMP Light client.

Task	Command
Display test session information on the TWAMP Light client.	**display nqa twamp-light client** [**test-session** *session-id* | **verbose**]
Display test session statistics on the TWAMP Light client, including two-way delay, two-way jitter, and two-way packet loss.	**display nqa twamp-light client statistics** { **two-way-delay** | **two-way-loss** } **test-session** *session-id*
Display the current monitoring results of reaction entries for the TWAMP Light test sessions.	**display nqa twamp-light client test-session reaction counters** [*session-id* [*item-number*]]
Clear TWAMP Light test session statistics.	**reset nqa twamp-light statistics** { **all** | **test-session** *session-id* }

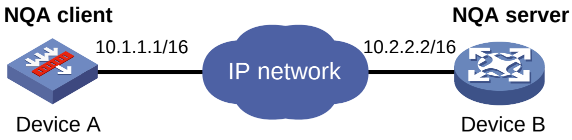
TWAMP Light configuration examples

Example: Configuring TWAMP Light test

Network configuration

As shown in Figure 3, configure a TWAMP Light test to measure the service quality from Device A to Device B.

Figure 3 Network diagram



Procedure

1. Assign IP addresses to interfaces, as shown in Figure 3. (Details not shown.)
2. Configure static routes or a routing protocol to make sure the devices can reach each other. (Details not shown.)
3. Configure Device B:

   # Enable the NQA server.

   ```
   <DeviceB> system-view
   [DeviceB] nqa server enable
   ```

   # Create test session **1** on the TWAMP Light responder with the destination IP address 10.2.2.2, source IP address 10.1.1.1, destination port 20000, and source port 10000.

   ```
   [DeviceB] nqa twamp-light responder
   [DeviceB-twamp-light-responder] test-session 1 ip destination 10.2.2.2 source
   10.1.1.1 destination-port 20000 source-port 10000
   [DeviceB-twamp-light-responder] quit
   ```
4. Configure Device A:

   # Create test session **1** on the TWAMP Light client.

   ```
   <DeviceA> system-view
   ```

```
[DeviceA] nqa twamp-light client
[DeviceA-nqa-twamp-light-client] test-session 1
```

# Specify 10.1.1.1 as the source IP address for the probe packets.

```
[DeviceA-nqa-twamp-light-client-session1] source ip 10.1.1.1
```

# Specify 10.2.2.2 as the destination IP address for the probe packets.

```
[DeviceA-nqa-twamp-light-client-session1] destination ip 10.2.2.2
```

# Specify 10000 as the source port number for the probe packets.

```
[DeviceA-nqa-twamp-light-client-session1] source port 10000
```

# Specify 20000 as the destination port number for the probe packets.

```
[DeviceA-nqa-twamp-light-client-session1] destination port 20000
```

# Create a TWAMP Light sender and enter its view.

```
<DeviceA> system-view
[DeviceA] nqa twamp-light sender
```

# Start the TWAMP Light test with the packet sending interval, statistics collection interval, and monitoring time set to 100, 10000, and 20000 in milliseconds, respectively.

```
<DeviceA> system-view
[DeviceA] nqa twamp-light sender
[DeviceA-nqa-twamp-light-sender] start test-session 1 permanent tx-interval 100
statistics-interval 10000 monitor-time 20000
[DeviceA-nqa-twamp-light-sender] quit
```

Verifying the configuration

# Display test session information on the TWAMP Light client.

```
[DeviceA-nqa-twamp-light-sender] display nqa twamp-light client
Brief information about all test sessions:
Total sessions: 1
Active sessions: 1
------------------------------------------------------------------------------------
ID    Status     Source IP/Port         Destination IP/Port
1     Active     10.1.1.1/10000         10.2.2.2/20000
```

# Display the test session statistics about the two-way packet loss for the test session **1**.

```
[DeviceA-nqa-twamp-light-sender] display nqa twamp-light client statistics two-way-loss
test-session 1
Latest two-way loss statistics:
    Index      Loss count      Loss ratio      Error count  Error ratio
    11006      5               50.0000%        0            0.0000%
    11007      3               30.0000%        0            0.0000%
    11008      4               40.0000%        0            0.0000%
    11009      8               80.0000%        0            0.0000%
--------------------------------------------------------------------
Average loss count :          5      Average loss ratio :  55.3333%
Maximum loss count :         10      Maximum loss ratio : 100.0000%
Minimum loss count :          1      Minimum loss ratio :  10.0000%
Average error count:          0      Average error ratio:   0.0000%
Maximum error count:          0      Maximum error ratio:   0.0000%
Minimum error count:          0      Minimum error ratio:   0.0000%
```

Contents

Configuring Track

About Track

The Track module works between application modules and detection modules. It shields the differences between various detection modules from application modules.

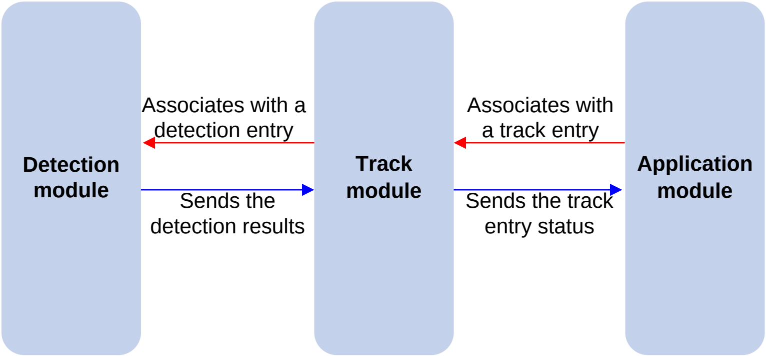
Collaboration mechanism

The Track module collaborates with detection modules and application modules.

As shown in Figure 1, collaboration is enabled when you associate the Track module with a detection module and an application module, and it operates as follows:

1. The detection module probes specific objects such as interface status, link status, network reachability, and network performance, and informs the Track module of detection results.
2. The Track module sends the detection results to the application module.
3. When notified of changes for the tracked object, the application modules can react to avoid communication interruption and network performance degradation.

Figure 1 Collaboration through the Track module


Detection module
Associates with a detection entry
Sends the detection results
Track module
Associates with a track entry
Sends the track entry status
Application module

Collaboration between the Track module and a detection module

The detection module sends the detection result of the tracked object to the Track module. The Track module changes the status of the track entry as follows:

- If the tracked object operates correctly, the state of the track entry is Positive. For example, the track entry state is Positive in one of the following conditions:
  - The target interface is up.
  - The target network is reachable.
- If the tracked object does not operate correctly, the state of the track entry is Negative. For example, the track entry state is Negative in one of the following conditions:
  - The target interface is down.
  - The target network is unreachable.
- If the detection result is invalid, the state of the track entry is NotReady. For example, the track entry state is NotReady if its associated NQA operation does not exist.

Collaboration between the Track module and an application module

The track module reports the track entry status changes to the application module. The application module can then take correct actions to avoid communication interruption and network performance degradation.

Supported detection modules

The following detection modules can be associated with the Track module:

- NQA.
- BFD.
- Interface management.
- Route management.

You can associate a track entry with an object of a detection module, such as the state of an interface or reachability of an IP route. The state of the track entry is determined by the state of the tracked object.

You can also associate a track entry with a list of objects called a tracked list. The state of a tracked list is determined by the states of all objects in the list. The following types of tracked lists are supported:

- **Boolean AND list**—The state of a Boolean AND list is determined by the states of the tracked objects using the Boolean AND operation.
- **Boolean OR list**—The state of a Boolean OR list is determined by the states of the tracked objects using the Boolean OR operation.
- **Percentage threshold list**—The state of a percentage threshold list is determined by comparing the percentage of Positive and Negative objects in the list with the percentage thresholds configured for the list.
- **Weight threshold list**—The state of a weight threshold list is determined by comparing the weight of Positive and Negative objects in the list with the weight thresholds configured for the list.

Supported application modules

The following application modules can be associated with the Track module:

- Static routing.
- PBR.
- Interface backup.
- Redundancy group.
- EAA.
- Security policy.

Restrictions and guidelines: Track configuration

When configuring a track entry for an application module, you can set a notification delay to avoid immediate notification of status changes.

When the delay is not configured and the route convergence is slower than the link state change notification, communication failures occur.

Collaboration application example

The following is an example of collaboration between NQA, Track, and static routing.

Configure a static route with next hop 192.168.0.88 on the device. If the next hop is reachable, the static route is valid. If the next hop becomes unreachable, the static route is invalid. For this purpose, configure NQA-Track-static routing collaboration as follows:

1. Create an NQA operation to monitor the accessibility of IP address 192.168.0.88.
2. Create a track entry and associate it with the NQA operation.
   - When next hop 192.168.0.88 is reachable, NQA sends the result to the Track module. The Track module sets the track entry to Positive state.
   - When the next hop becomes unreachable, NQA sends the result to the Track module. The Track module sets the track entry to Negative state.
3. Associate the track entry with the static route.
   - When the track entry is in Positive state, the static routing module considers the static route to be valid.
   - When the track entry is in Negative state, the static routing module considers the static route to be invalid.

Track tasks at a glance

To implement the collaboration function, establish associations between the Track module and detection modules, and between the Track module and application modules.

To configure the Track module, perform the following tasks:

1. Associating Track with a detection module object
   - Associating Track with NQA
   - Associating Track with BFD
   - Associating Track with interface management
   - Associating Track with route management
2. Associating Track with a tracked list
   - Associating Track with a Boolean list
   - Associating Track with a percentage threshold list
   - Associating Track with a weight threshold list
3. Associating the Track module with an application module
   - Associating Track with static routing
   - Associating Track with PBR
   - Associating Track with interface backup
   - Associating Track with the redundancy group module
   - Associating Track with EAA
   - Associating Track with a security policy rule

Associating Track with a detection module object

Associating Track with NQA

About this task

NQA supports multiple operation types to analyze network performance and service quality. For example, an NQA operation can periodically detect whether a destination is reachable, or whether a TCP connection can be established.

An NQA operation operates as follows when it is associated with a track entry:

- If the consecutive probe failures reach the specified threshold, the NQA module notifies the Track module that the tracked object has malfunctioned. The Track module then sets the track entry to Negative state.

- If the specified threshold is not reached, the NQA module notifies the Track module that the tracked object is operating correctly. The Track module then sets the track entry to Positive state.

For more information about NQA, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

If you associate a track entry with a nonexistent NQA operation or reaction entry, the state of the track entry is NotReady.

Procedure

1. Enter system view.

   `system-view`

2. Create a track entry, associate it with an NQA reaction entry, and enter its view.

   track *track-entry-number* **nqa entry** *admin-name operation-tag* **reaction** *item-number*

3. Set the delay for notifying the application module of track entry state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the track entry state changes.

Associating Track with BFD

About this task

BFD supports the control packet mode and echo packet mode. A track entry can be associated only with the echo-mode BFD session. For more information about BFD, see *Network Management and Monitoring Configuration Guide*.

The associated Track and BFD operate as follows:

- If the BFD detects that the link fails, it informs the Track module of the link failure. The Track module sets the track entry to Negative state.
- If the BFD detects that the link is operating correctly, the Track module sets the track entry to Positive state.

Restrictions and guidelines

When you associate a track entry with BFD, do not configure the virtual IP address of a VRRP group as the local or remote address of the BFD session.

Prerequisites

Before you associate Track with BFD, configure the source IP address of BFD echo packets. For more information, see BFD configuration in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Create a track entry, associate it with a BFD session, and enter its view.

   track *track-entry-number* **bfd echo interface** *interface-type interface-number* **remote ip** *remote-ip-address* **local ip** *local-ip-address*

3. Set the delay for notifying the application module of track entry state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

By default, the Track module notifies the application module immediately when the track entry state changes.

Associating Track with interface management

About this task

The interface management module monitors the link status, physical status, or network-layer protocol status of interfaces. The associated Track and interface management operate as follows:

- When the link status, physical status, or network-layer protocol status of the interface changes to up, the interface management module informs the Track module of the change. The Track module sets the track entry to Positive state.
- When the link status, physical status, or network-layer protocol status of the interface changes to down, the interface management module informs the Track module of the change. The Track module sets the track entry to Negative state.

Procedure

1. Enter system view.

   `system-view`

2. Create a track entry, associate it with an interface, and enter its view.
   - Create a track entry to monitor the link status of an interface.

     track *track-entry-number* **interface** *interface-type interface-number*
   - Create a track entry to monitor the physical status of an interface.

     track *track-entry-number* **interface** *interface-type interface-number* **physical**
   - Create a track entry to monitor the network layer protocol status of an interface.

     track *track-entry-number* **interface** *interface-type interface-number* **protocol** { **ipv4** | **ipv6** }

3. Set the delay for notifying the application module of track entry state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the track entry state changes.

Associating Track with route management

About this task

The route management module monitors route entry changes in the routing table. The associated Track and route management operate as follows:

- When a monitored route entry is found in the routing table, the route management module informs the Track module. The Track module sets the track entry to Positive state.
- When a monitored route entry is removed from the routing table, the route management module informs the Track module of the change. The Track module sets the track entry to Negative state.

Procedure

1. Enter system view.

   `system-view`

2. Create a track entry, associate it with an IP route, and enter its view.

track *track-entry-number* **ip route** [**vpn-instance** *vpn-instance-name*] **ip-address** { *mask-length* | *mask* } **reachability**

3. Set the delay for notifying the application module of track entry state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the track entry state changes.

Associating Track with a tracked list

Associating Track with a Boolean list

About this task

A Boolean list is a list of tracked objects based on a Boolean logic. It can be further divided into the following types:

- **Boolean AND list**—A Boolean AND list is set to the Positive state only when all objects are in Positive state. If one or more objects are in Negative state, the tracked list is set to the Negative state.
- **Boolean OR list**—A Boolean OR list is set to the Positive state if any object is in Positive state. If all objects are in Negative state, the tracked list is set to the Negative state.

Procedure

1. Enter system view.

   system-view

2. Create a track entry.

   See "Associating Track with a detection module object."

   Create a track entry before you add it as a tracked object to a tracked list.

   A minimum of one track entry must be created.

3. Create a Boolean tracked list and enter its view.

   track *track-entry-number* **list boolean** { **and** | **or** }

4. Add the track entry as an object to the tracked list.

   object *track-entry-number* [**not**]

   Repeat this step to add all interested objects to the tracked list.

5. (Optional.) Set the delay for notifying the application module of tracked list state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the tracked list state changes.

Associating Track with a percentage threshold list

About this task

A percentage threshold list uses a percentage threshold to measure the state of the list.

- If the percentage of Positive objects is equal to or above the positive state threshold, the list is set to the Positive state.
- If the percentage of Positive objects is equal to or below the negative state threshold, the list is set to the Negative state.
- The state of a percentage threshold list remains unchanged if the percentage of Positive objects is below the positive state threshold and above the negative state threshold.

Procedure

1. Enter system view.

   system-view

2. Create a track entry.

   See "Associating Track with a detection module object."

   Create a track entry before you add it as an tracked object to a tracked list.

   A minimum of one track entry must be created.

3. Create a percentage threshold list and enter its view.

   track *track-entry-number* **list threshold percentage**

4. Add the track entry as an object to the tracked list.

   object *track-entry-number*

   Repeat this step to add all interested objects to the tracked list.

5. Configure the threshold values used to determine the state of the percentage threshold list.

   threshold percentage { **negative** *negative-threshold* | **positive** *positive-threshold* } *

   By default, the negative state threshold is 0% and the positive state threshold is 1%.

6. (Optional.) Set the delay for notifying the application module of tracked list state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the tracked list state changes.

Associating Track with a weight threshold list

About this task

A weight threshold list uses a weight threshold to measure the state of the list.

- If the total weight of Positive objects is equal to or above the positive state threshold, the list is set to the Positive state.
- If the total weight of Positive objects is equal to or below the negative state threshold, the list is set to the Negative state.
- The state of a weight threshold list remains unchanged if the total weight of Positive objects is below the positive state threshold and above the negative state threshold.

Procedure

1. Enter system view.

   system-view

2. Create a track entry.

   See "Associating Track with a detection module object."

   Create a track entry before you add it as an tracked object to a tracked list.

   A minimum of one track entry must be created.

3. Create a weight threshold list and enter its view.

   track *track-entry-number* **list threshold weight**

4. Add the track entry as an object to the tracked list.

   object *track-entry-number* [**weight** *weight*]

   Repeat this step to add all interested objects to the tracked list.

5. Configure the threshold values used to determine the state of the weight threshold list.

threshold weight { **negative** *negative-threshold* | **positive** *positive-threshold* } *

By default, the negative state threshold is 0 and the positive state threshold is 1.

6. (Optional.) Set the delay for notifying the application module of tracked list state changes.

   delay { **negative** *negative-time* | **positive** *positive-time* } *

   By default, the Track module notifies the application module immediately when the tracked list state changes.

Associating the Track module with an application module

Before you associate the Track module with an application module, make sure the associated track entry has been created.

Prerequisites for associating the Track module with an application module

Create a track entry first before you associate it with an application module.

An application module might obtain incorrect track entry status information if the associated track entry does not exist.

Associating Track with static routing

About this task

A static route is a manually configured route to route packets. For more information about static route configuration, see *Layer 3—IP Routing Configuration Guide*.

Static routes cannot adapt to network topology changes. Link failures or network topological changes can make the routes unreachable and cause communication interruption.

To resolve this problem, configure another route to back up the static route. When the static route is reachable, packets are forwarded through the static route. When the static route is unreachable, packets are forwarded through the backup route.

To check the accessibility of a static route in real time, associate the Track module with the static route.

If you specify the next hop but not the output interface when configuring a static route, you can configure the static routing-Track-detection module collaboration. This collaboration enables you to verify the accessibility of the static route based on the track entry state.

- If the track entry is in Positive state, the following conditions exist:
  - The next hop of the static route is reachable.
  - The configured static route is valid.
- If the track entry is in Negative state, the following conditions exist:
  - The next hop of the static route is not reachable.
  - The configured static route is invalid.
- If the track entry is in NotReady state, the following conditions exist:
  - The accessibility of the next hop of the static route is unknown.
  - The static route is valid.

Restrictions and guidelines

In static routing-Track-NQA collaboration, you must configure the same VPN instance name for the NQA operation and the next hop of the static route. Otherwise, the accessibility detection cannot operate correctly.

If a static route needs route recursion, the associated track entry must monitor the next hop of the recursive route. The next hop of the static route cannot be monitored. Otherwise, a valid route might be considered invalid.

Associating Track with an IPv4 static route

1. Enter system view.

   ```
   system-view
   ```

2. Associate an IPv4 static route with a track entry to check the accessibility of the next hop.

   Public network:

   ```
   ip route-static { dest-address { mask-length | mask } | group group-name }
   { interface-type interface-number [ next-hop-address ] track
   track-entry-number | next-hop-address track track-entry-number |
   vpn-instance d-vpn-instance-name next-hop-address track
   track-entry-number } [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   VPN:

   ```
   ip route-static vpn-instance s-vpn-instance-name { dest-address
   { mask-length | mask } | group group-name } { interface-type
   interface-number [ next-hop-address ] track track-entry-number |
   next-hop-address [ public ] track track-entry-number | vpn-instance
   d-vpn-instance-name next-hop-address track track-entry-number }
   [ preference preference ] [ tag tag-value ] [ description text ]
   ```

   By default, Track is not associated with any IPv4 static routes.

Associating Track with an IPv6 static route

1. Enter system view.

   ```
   system-view
   ```

2. Associate an IPv6 static route with a track entry to check the accessibility of the next hop.

   Public network:

   ```
   ipv6 route-static ipv6-address prefix-length { interface-type
   interface-number [ next-hop-address ] track track-entry-number |
   [ vpn-instance d-vpn-instance-name ] next-hop-address track
   track-entry-number } [ preference preference ] [ tag tag-value ]
   [ description text ]
   ```

   VPN:

   ```
   ipv6 route-static vpn-instance s-vpn-instance-name ipv6-address
   prefix-length { interface-type interface-number [ next-hop-address ]
   track track-entry-number | vpn-instance d-vpn-instance-name
   next-hop-address track track-entry-number } [ preference preference ]
   [ tag tag-value ] [ description text ]
   ```

   By default, Track is not associated with any IPv6 static routes.

Associating Track with PBR

About this task

PBR uses user-defined policies to route packets. You can specify parameters in a PBR policy to guide the forwarding of the packets that match specific criteria. For more information about PBR, see *Layer 3—IP Routing Configuration Guide*.

PBR cannot detect the availability of any action taken on packets. When an action is not available, packets processed by the action might be discarded. For example, if the output interface specified for PBR fails, PBR cannot detect the failure, and continues to forward matching packets out of the interface.

To enable PBR to detect topology changes and improve the flexibility of the PBR application, configure Track-PBR-detection module collaboration.

After you associate a track entry with an apply clause, the detection module associated with the track entry sends Track the detection result of the availability of the tracked object.

- The Positive state of the track entry indicates that the object is available, and the apply clause is valid.
- The Negative state of the track entry indicates that the object is not available, and the apply clause is invalid.
- The NotReady state of the track entry indicates that the apply clause is valid.

The following objects can be associated with a track entry:

- Output interface.
- Next hop.
- Default output interface.
- Default next hop.

Prerequisites for Track association with PBR

Before you associate Track with PBR, create a policy node, and set the match criteria.

Associating Track with PBR

1. Enter system view.

   system-view

2. Create a policy node and enter its view.

   policy-based-route *policy-name* [**deny** | **permit**] **node** *node-number*

3. Set match criteria. Choose the options to configure as needed:

   - Set an ACL match criterion.

     if-match acl { *acl-number* | **name** *acl-name* }

     By default, no ACL match criterion is set.

     The ACL match criterion cannot match Layer 2 information.

   - Set a packet length match criterion.

     if-match packet-length *min-len max-len*

     By default, no packet length match criterion is set.

4. Set actions and associate the policy node with a track entry. Choose the options to configure as needed:

   - Set the output interface.

     apply output-interface { *interface-type interface-number* [**track** *track-entry-number*] }&<1-4>

     By default, no output interface is set.

- Set the next hop.

  ```
  apply next-hop [ vpn-instance vpn-instance-name | inbound-vpn ]
  { ip-address [ direct ] [ track track-entry-number ] [ weight
  weight-value ] }&<1-4>
  ```

  By default, no next hop is set.

- Set the default output interface.

  ```
  apply default-output-interface { interface-type interface-number
  [ track track-entry-number ] }&<1-4>
  ```

  By default, no default output interface is set.

- Set the default next hop.

  ```
  apply default-next-hop [ vpn-instance vpn-instance-name |
  inbound-vpn ] { ip-address [ direct ] [ track
  track-entry-number ] }&<1-4>
  ```

  By default, no default next hop is set.

Associating Track with IPv6 PBR

1. Enter system view.

   ```
   system-view
   ```

2. Create an IPv6 policy node and enter its view.

   ```
   ipv6 policy-based-route policy-name [ deny | permit ] node node-number
   ```

3. Set match criteria. Choose the options to configure as needed:

   - Set an ACL match criterion.

     ```
     if-match acl { ipv6-acl-number | name ipv6-acl-name }
     ```

     By default, no ACL match criterion is set.

     The ACL match criterion cannot match Layer 2 information.

   - Set an IPv6 packet length match criterion.

     ```
     if-match packet-length min-len max-len
     ```

     By default, no packet length match criterion is set.

4. Set actions and associate the policy node with a track entry. Choose the options to configure as needed:

   - Set the output interface.

     ```
     apply output-interface { interface-type interface-number [ track
     track-entry-number ] }&<1-4>
     ```

     By default, no output interface is set.

   - Set the next hop.

     ```
     apply next-hop [ vpn-instance vpn-instance-name | inbound-vpn ]
     { ipv6-address [ direct ] [ track track-entry-number ] [ weight
     weight-value ] } &<1-4>
     ```

     By default, no next hop is set.

   - Set the default output interface.

     ```
     apply default-output-interface { interface-type interface-number
     [ track track-entry-number ] }&<1-4>
     ```

     By default, no default output interface is set.

   - Set the default next hop.

     ```
     apply default-next-hop [ vpn-instance vpn-instance-name |
     inbound-vpn ] { ipv6-address [ direct ] [ track
     track-entry-number ] }&<1-4>
     ```

By default, no default next hop is set.

Associating Track with interface backup

About this task

To enable a standby interface to detect the status of the active interface, you can associate the standby interface with a track entry.

- If the track entry is in Positive state, the following conditions exist:
  - The link where the active interface resides operates correctly.
  - The standby interfaces stay in backup state.
- If the track entry is in Negative state, the following conditions exist:
  - The link where the active interface resides has failed.
  - A standby interface changes to the active interface for data transmission.
- If the track entry is in always NotReady state, the following conditions exist:
  - The association does not take effect.
  - Each interface keeps its original forwarding state.

  When the track entry turns to NotReady from other state, a standby interface becomes the active interface.

For more information about configuring interface backup, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Associate the interface with a track entry.

   `backup track` *`track-entry-number`*

   By default, no track entry is associated with an interface.

   You can associate an interface with only one track entry.

   If you execute this command multiple times, the most recent configuration takes effect.

Associating Track with the redundancy group module

About this task

The redundancy group can fast detect the link and interface failures after you associate it with Track.

Track changes the track entry state based on the monitoring result of a detection module, and notifies the track entry state change to the redundancy group.

- If the track entry state changes to Positive, the system increases the weight value of the redundancy group node. When the value of the redundancy group is greater than 0, the node can operate correctly.
- If the track entry state changes to Negative or NotReady, the system reduces the weight value of redundancy group node. If the value of the redundancy group is less than 0, the node cannot operate correctly. A node switchover occurs. The members (including Reth interfaces) on the other node take over.

For more information about redundancy groups, see *Virtual Technologies Configuration Guide*.

Restrictions and guidelines

To associate Track with a redundancy group configured with automatic node switchover, you must specify the **interface** *interface-type interface-number* option in the **track** command. When the specified interface fails, it will not be shut down by the Reth module.

Procedure

1. Enter system view.

   system-view

2. Create a redundancy group and enter its view.

   redundancy group *group-name*

3. Create a redundancy group node and enter its view.

   node *node-id*

4. Associate Track with the redundancy group.

   track *track-entry-number* [**reduced** *weight-reduced*] [**interface** *interface-type interface-number*]

   By default, no track entry is associated with a redundancy group.

Associating Track with EAA

About this task

You can configure EAA track event monitor policies to monitor the positive-to-negative or negative-to-positive state changes of track entries.

- If you specify only one track entry for a policy, EAA triggers the policy when it detects the specified state change on the track entry.
- If you specify multiple track entries for a policy, EAA triggers the policy when it detects the specified state change on the last monitored track entry. For example, if you configure a policy to monitor the positive-to-negative state change of multiple track entries, EAA triggers the policy when the last positive track entry monitored by the policy is changed to the Negative state.

You can set a suppression time for a track event monitor policy. The timer starts when the policy is triggered. The system does not process messages that report the monitored track event until the timer times out.

For more information about EAA, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Create a CLI-defined monitor policy and enter its view, or enter the view of an existing CLI-defined monitor policy.

   rtm cli-policy *policy-name*

3. Configure a track event.

   event track *track-entry-number-list* **state** { **negative** | **positive** } [**suppress-time** *suppress-time*]

   By default, a monitor policy does not monitor any track event.

Associating Track with a security policy rule

About this task

Perform this task to enable the collaboration between Track and a security policy rule. The collaboration operates as follows:

- If the rule is associated with the Negative state of a track entry, the device takes the following actions:
  - Sets the rule state to Active if the track entry is in Negative state.
  - Sets the rule state to Inactive if the track entry is in Positive state.
- If the rule is associated with the Positive state of a track entry, the device takes the following actions:
  - Sets the rule state to Active if the track entry is in Positive state.
  - Sets the rule state to Inactive if the track entry is in Negative state.

Procedure

1. Enter system view.
   system-view
2. Enter IPv4 or IPv6 security policy view.
   security-policy { **ip** | **ipv6** }
3. Enter security policy rule view.
   rule { *rule-id* | **name** *name* } *
4. Associate the rule with a track entry.
   track { **negative** | **positive** } *track-entry-number*
   By default, no track entry is associated with a security policy rule.

Display and maintenance commands for Track

Execute **display** commands in any view.

Task	Command
Display information about track entries.	**display track** { *track-entry-number* | **all** [**negative** | **positive**] } [**brief**]

Track configuration examples

Example: Configuring static routing-Track-NQA collaboration

Network configuration

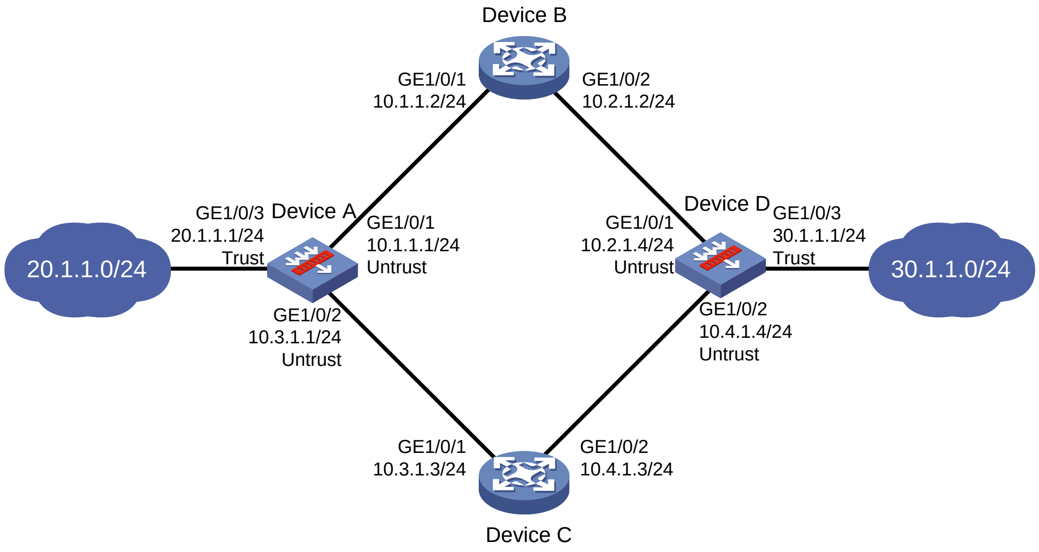
As shown in Figure 2:

- Device A is the default gateway of the hosts in network 20.1.1.0/24.
- Device D is the default gateway of the hosts in network 30.1.1.0/24.
- Hosts in the two networks communicate with each other through static routes.

To ensure network availability, configure route backup and static routing-Track-NQA collaboration on Device A and Device D as follows:

- On Device A, assign a higher priority to the static route to 30.1.1.0/24 with next hop Device B. This route is the master route. The static route to 30.1.1.0/24 with next hop Device C acts as the backup route. When the master route is unavailable, the backup route takes effect.
- On Device D, assign a higher priority to the static route to 20.1.1.0/24 with next hop Device B. This route is the master route. The static route to 20.1.1.0/24 with next hop Device C acts as the backup route. When the master route is unavailable, the backup route takes effect.

Figure 2 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

   # Configure a main static route to 30.1.1.0/24 with next hop 10.1.1.2 and default priority 60, and associate the static route with track entry 1.

   ```
   [DeviceA] ip route-static 30.1.1.0 24 10.1.1.2 track 1
   ```

   # Configure a backup static route to 30.1.1.0/24 with next hop 10.3.1.3 and priority 80.

   ```
   [DeviceA] ip route-static 30.1.1.0 24 10.3.1.3 preference 80
   ```

   # Configure a static route to 10.2.1.4/24 with next hop 10.1.1.2 and default priority 60. This static route will be used in an NQA operation.

   ```
   [DeviceA] ip route-static 10.2.1.4 24 10.1.1.2
   ```

3. Add interfaces to security zones.

   ```
   [DeviceA] security-zone name untrust
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
   [DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
   [DeviceA-security-zone-Untrust] quit
   [DeviceA] security-zone name trust
   [DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/3
   [DeviceA-security-zone-Trust] quit
   ```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit packets from network 20.1.1.0/24 to network 30.1.1.0/24.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] destination-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit packets from network 30.1.1.0/24 to network 20.1.1.0/24.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-2-untrust-trust] source-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] destination-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] action pass
[DeviceA-security-policy-ip-2-untrust-trust] quit
```

   # Configure a rule named **nqalocalout** to allow Device A to send NQA probe packets to Device D.

```
[DeviceA-security-policy-ip] rule name nqalocalout
[DeviceA-security-policy-ip-3-nqalocalout] source-zone local
[DeviceA-security-policy-ip-3-nqalocalout] destination-zone untrust
[DeviceA-security-policy-ip-2-nqalocalout] service ping
[DeviceA-security-policy-ip-3-nqalocalout] action pass
[DeviceA-security-policy-ip-3-nqalocalout] quit
```

   # Configure a rule named **nqalocalin** to allow Device A to receive the NQA probe packets from Device D.

```
[DeviceA-security-policy-ip] rule name nqalocalin
[DeviceA-security-policy-ip-4-nqalocalin] source-zone untrust
[DeviceA-security-policy-ip-4-nqalocalin] destination-zone local
[DeviceA-security-policy-ip-4-nqalocalin] service ping
[DeviceA-security-policy-ip-4-nqalocalin] action pass
[DeviceA-security-policy-ip-4-nqalocalin] quit
[DeviceA-security-policy-ip] quit
```

5. Create an NQA operation to test connectivity of path Device A—Device B—Device D.

```
[DeviceA] nqa entry admin test
[DeviceA-nqa-admin-test] type icmp-echo
[DeviceA-nqa-admin-test-icmp-echo] destination ip 10.2.1.4
[DeviceA-nqa-admin-test-icmp-echo] next-hop ip 10.1.1.2
[DeviceA-nqa-admin-test-icmp-echo] frequency 100
[DeviceA-nqa-admin-test-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 5 action-type trigger-only
[DeviceA-nqa-admin-test-icmp-echo] quit
[DeviceA] nqa schedule admin test start-time now lifetime forever
```

6. Associate track entry 1 with reaction entry 1 of the NQA operation.

```
[DeviceA] track 1 nqa entry admin test reaction 1
```

```
[DeviceA-track-1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address 10.1.1.2 255.255.255.0
   [DeviceB-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

   # Configure a static route to 30.1.1.0/24 with next hop 10.2.1.4.

   ```
   [DeviceB] ip route-static 30.1.1.0 24 10.2.1.4
   ```

   # Configure a static route to 20.1.1.0/24 with next hop 10.1.1.1.

   ```
   [DeviceB] ip route-static 20.1.1.0 24 10.1.1.1
   ```

Configuring Device C

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceC> system-view
   [DeviceC] interface gigabitethernet 1/0/1
   [DeviceC-GigabitEthernet1/0/1] ip address 10.3.1.3 255.255.255.0
   [DeviceC-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

   # Configure a static route to 30.1.1.0/24 with next hop 10.4.1.4.

   ```
   [DeviceC] ip route-static 30.1.1.0 24 10.4.1.4
   ```

   # Configure a static route to 20.1.1.0/24 with next hop 10.3.1.1.

   ```
   [DeviceC] ip route-static 20.1.1.0 24 10.3.1.1
   ```

Configuring Device D

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceD> system-view
   [DeviceD] interface gigabitethernet 1/0/1
   [DeviceD-GigabitEthernet1/0/1] ip address 10.2.1.4 255.255.255.0
   [DeviceD-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

   # Configure a main static route to 20.1.1.0/24 with next hop 10.2.1.2 and default priority 60, and associate the static route with track entry 1.

   ```
   [DeviceD] ip route-static 20.1.1.0 24 10.2.1.2 track 1
   ```

   # Configure a backup static route to 20.1.1.0/24 with next hop 10.4.1.3 and priority 80.

   ```
   [DeviceD] ip route-static 20.1.1.0 24 10.4.1.3 preference 80
   ```

   # Configure a static route to 10.1.1.1/24 with next hop 10.2.1.2 and default priority 60. This static route will be used in an NQA operation.

   ```
   [DeviceD] ip route-static 10.1.1.1 24 10.2.1.2
   ```

3. Add interfaces to security zones.

```
[DeviceD] security-zone name untrust
[DeviceD-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceD-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceD-security-zone-Untrust] quit
[DeviceD] security-zone name trust
[DeviceD-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceD-security-zone-Trust] quit
```

4. Configure a security policy:

   # Configure a rule named **trust-untrust** to permit packets from network 30.1.1.0/24 to network 20.1.1.0/24.

```
[DeviceD] security-policy ip
[DeviceD-security-policy-ip] rule name trust-untrust
[DeviceD-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceD-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceD-security-policy-ip-1-trust-untrust] source-ip-subnet 30.1.1.0 24
[DeviceD-security-policy-ip-1-trust-untrust] destination-ip-subnet 20.1.1.0 24
[DeviceD-security-policy-ip-1-trust-untrust] action pass
[DeviceD-security-policy-ip-1-trust-untrust] quit
```

   # Configure a rule named **untrust-trust** to permit packets from network 20.1.1.0/24 to network 30.1.1.0/24.

```
[DeviceD-security-policy-ip] rule name untrust-trust
[DeviceD-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceD-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceD-security-policy-ip-2-untrust-trust] source-ip-subnet 20.1.1.0 24
[DeviceD-security-policy-ip-2-untrust-trust] destination-ip-subnet 30.1.1.0 24
[DeviceD-security-policy-ip-2-untrust-trust] action pass
[DeviceD-security-policy-ip-2-untrust-trust] quit
```

   # Configure a rule named **nqalocalout** to allow Device D to send NQA probe packets to Device A.

```
[DeviceD-security-policy-ip] rule name nqalocalout
[DeviceD-security-policy-ip-3-nqalocalout] source-zone local
[DeviceD-security-policy-ip-3-nqalocalout] destination-zone untrust
[DeviceD-security-policy-ip-3-nqalocalout] service ping
[DeviceD-security-policy-ip-3-nqalocalout] action pass
[DeviceD-security-policy-ip-3-nqalocalout] quit
```

   # Configure a rule named **nqalocalin** to allow Device D to receive the NQA probe packets from Device A.

```
[DeviceD-security-policy-ip] rule name nqalocalin
[DeviceD-security-policy-ip-4-nqalocalin] source-zone untrust
[DeviceD-security-policy-ip-4-nqalocalin] destination-zone local
[DeviceD-security-policy-ip-4-nqalocalin] service ping
[DeviceD-security-policy-ip-4-nqalocalin] action pass
[DeviceD-security-policy-ip-4-nqalocalin] quit
[DeviceD-security-policy-ip] quit
```

5. Create an NQA operation to test connectivity of path Device D—Device B—Device A.

```
[DeviceD] nqa entry admin test
[DeviceD-nqa-admin-test] type icmp-echo
[DeviceD-nqa-admin-test-icmp-echo] destination ip 10.1.1.1
[DeviceD-nqa-admin-test-icmp-echo] next-hop ip 10.2.1.2
```

```
[DeviceD-nqa-admin-test-icmp-echo] frequency 100
[DeviceD-nqa-admin-test-icmp-echo] reaction 1 checked-element probe-fail
threshold-type consecutive 5 action-type trigger-only
[DeviceD-nqa-admin-test-icmp-echo] quit
[DeviceD] nqa schedule admin test start-time now lifetime forever
```

6. Associate track entry 1 with reaction entry 1 of the NQA operation.

```
[DeviceD] track 1 nqa entry admin test reaction 1
[DeviceD-track-1] quit
```

Verifying the configuration

# Display track entry information on Device A.

```
[DeviceA] display track all
Track ID: 1
  State: Positive
  Duration: 0 days 0 hours 0 minutes 32 seconds
  Tracked object type: NQA
  Notification delay: Positive 0, Negative 0 (in seconds)
  Tracked object:
    NQA entry: admin test
    Reaction: 1
    Remote IP/URL: 10.2.1.4
    Local IP:--
    Interface:--
```

The output shows that the status of track entry 1 is Positive, indicating that the NQA operation has succeeded and the master route is available.

# Display the routing table of Device A.

```
[DeviceA] display ip routing-table

Destinations : 10       Routes : 10

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.1.1.0/24         Direct 0    0            10.1.1.1        GE1/0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
10.2.1.0/24         Static 60   0            10.1.1.2        GE1/0/1
10.3.1.0/24         Direct 0    0            10.3.1.1        GE1/0/2
10.3.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         Direct 0    0            20.1.1.1        GE1/0/3
20.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
30.1.1.0/24         Static 60   0            10.1.1.2        GE1/0/1
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

The output shows that Device A forwards packets to 30.1.1.0/24 through Device B.

# Remove the IP address of GigabitEthernet 1/0/1 on Device B.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] undo ip address
```

# Display information about the track entry on Device A.

```
[DeviceA] display track all
```

```
Track ID: 1
  State: Negative
  Duration: 0 days 0 hours 0 minutes 32 seconds
  Tracked object type: NQA
  Notification delay: Positive 0, Negative 0 (in seconds)
  Tracked object:
    NQA entry: admin test
    Reaction: 1
    Remote IP/URL: 10.2.1.4
    Local IP:--
    Interface:--
```

The output shows that the status of the track entry is Negative, indicating that the NQA operation has failed and the master route is unavailable.

# Display the routing table of Device A.

```
[DeviceA] display ip routing-table

Destinations : 10       Routes : 10

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.1.1.0/24         Direct 0    0            10.1.1.1        GE1/0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
10.2.1.0/24         Static 60   0            10.1.1.2        GE1/0/1
10.3.1.0/24         Direct 0    0            10.3.1.1        GE1/0/2
10.3.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         Direct 0    0            20.1.1.1        GE1/0/3
20.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
30.1.1.0/24         Static 80   0            10.3.1.3        GE1/0/2
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

The output shows that Device A forwards packets to 30.1.1.0/24 through Device C. The backup static route has taken effect.

# Verify that hosts in 20.1.1.0/24 can communicate with the hosts in 30.1.1.0/24 when the master route fails.

```
[DeviceA] ping -a 20.1.1.1 30.1.1.1
Ping 30.1.1.1: 56  data bytes, press CTRL_C to break
Reply from 30.1.1.1: bytes=56 Sequence=1 ttl=254 time=2 ms
Reply from 30.1.1.1: bytes=56 Sequence=2 ttl=254 time=1 ms
Reply from 30.1.1.1: bytes=56 Sequence=3 ttl=254 time=1 ms
Reply from 30.1.1.1: bytes=56 Sequence=4 ttl=254 time=2 ms
Reply from 30.1.1.1: bytes=56 Sequence=5 ttl=254 time=1 ms
--- Ping statistics for 30.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.00% packet loss
round-trip min/avg/max/std-dev = 1/1/2/1 ms
```

# Verify that the hosts in 30.1.1.0/24 can communicate with the hosts in 20.1.1.0/24 when the master route fails.

```
[DeviceD] ping -a 30.1.1.1 20.1.1.1
Ping 20.1.1.1: 56  data bytes, press CTRL_C to break
Reply from 20.1.1.1: bytes=56 Sequence=1 ttl=254 time=2 ms
Reply from 20.1.1.1: bytes=56 Sequence=2 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=3 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=4 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=5 ttl=254 time=1 ms

--- Ping statistics for 20.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.00% packet loss
round-trip min/avg/max/std-dev = 1/1/2/1 ms
```

Example: Configuring static routing-Track-BFD collaboration

Network configuration

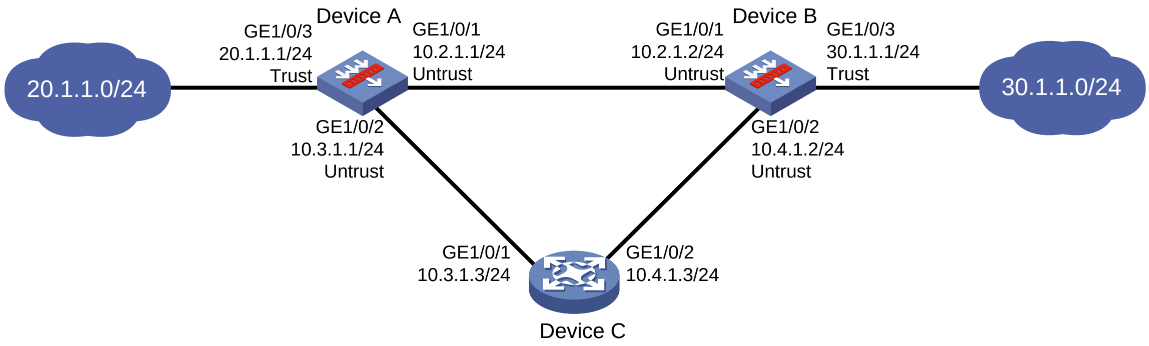
As shown in Figure 3:

- Device A is the default gateway of the hosts in network 20.1.1.0/24.
- Device B is the default gateway of the hosts in network 30.1.1.0/24.
- Hosts in the two networks communicate with each other through static routes.

To ensure network availability, configure route backup and static routing-Track-BFD collaboration on Device A and Device B as follows:

- On Device A, assign a higher priority to the static route to 30.1.1.0/24 with next hop Device B. This route is the master route. The static route to 30.1.1.0/24 with next hop Device C acts as the backup route. When the master route is unavailable, BFD can quickly detect the route failure to make the backup route take effect.
- On Device B, assign a higher priority to the static route to 20.1.1.0/24 with next hop Device A. This route is the master route. The static route to 20.1.1.0/24 with next hop Device C acts as the backup route. When the master route is unavailable, BFD can quickly detect the route failure to make the backup route take effect.

Figure 3 Network diagram



Configuring Device A

1. Assign IP addresses to interfaces:

   # Assign an IP address to interface GigabitEthernet 1/0/1.

   ```
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.2.1.1 255.255.255.0
   [DeviceA-GigabitEthernet1/0/1] quit
   ```

   # Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

# Configure a main static route to 30.1.1.0/24 with next hop 10.2.1.2 and default priority 60, and associate the static route with track entry 1.

```
[DeviceA] ip route-static 30.1.1.0 24 10.2.1.2 track 1
```

# Configure a backup static route to 30.1.1.0/24 with next hop 10.3.1.3 and priority 80.

```
[DeviceA] ip route-static 30.1.1.0 24 10.3.1.3 preference 80
```

3. Add interfaces to security zones.

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceA-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceA-security-zone-Trust] quit
```

4. Configure a security policy:

# Configure a rule named **trust-untrust** to permit packets from network 20.1.1.0/24 to network 30.1.1.0/24.

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] destination-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit packets from network 30.1.1.0/24 to network 20.1.1.0/24.

```
[DeviceA-security-policy-ip] rule name untrust-trust
[DeviceA-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceA-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceA-security-policy-ip-2-untrust-trust] source-ip-subnet 30.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] destination-ip-subnet 20.1.1.0 24
[DeviceA-security-policy-ip-2-untrust-trust] action pass
[DeviceA-security-policy-ip-2-untrust-trust] quit
```

# Configure a rule named **bfdlocalout** to allow Device A to send BFD echo packets to Device B.

```
[DeviceA-security-policy-ip] rule name bfdlocalout
[DeviceA-security-policy-ip-3-bfdlocalout] source-zone local
[DeviceA-security-policy-ip-3-bfdlocalout] destination-zone untrust
[DeviceA-security-policy-ip-2-bfdlocalout] service bfd-echo
[DeviceA-security-policy-ip-3-bfdlocalout] action pass
[DeviceA-security-policy-ip-3-bfdlocalout] quit
```

# Configure a rule named **bfdlocalin** to allow Device A to receive BFD echo packets.

```
[DeviceA-security-policy-ip] rule name bfdlocalin
[DeviceA-security-policy-ip-4-bfdlocalin] source-zone untrust
[DeviceA-security-policy-ip-4-bfdlocalin] destination-zone local
[DeviceA-security-policy-ip-2-bfdlocalin] service bfd-echo
[DeviceA-security-policy-ip-4-bfdlocalin] action pass
[DeviceA-security-policy-ip-4-bfdlocalin] quit
```

# Configure a rule named **untrust-untrust** to allow Device A to respond to the BFD echo packets from Device B.

```
[DeviceA-security-policy-ip] rule name untrust-untrust
[DeviceA-security-policy-ip-5-untrust-untrust] source-zone untrust
[DeviceA-security-policy-ip-5-untrust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-5-untrust-untrust] source-ip-host 1.1.1.1
[DeviceA-security-policy-ip-5-untrust-untrust] destination-ip-host 10.2.1.2
[DeviceA-security-policy-ip-5-untrust-untrust] action pass
[DeviceA-security-policy-ip-5-untrust-untrust] quit
```

# Configure a rule named **pinglocalout** to allow Device A to send ping packets to Device B.

```
[DeviceA-security-policy-ip] rule name pinglocalout
[DeviceA-security-policy-ip-6-pinglocalout] source-zone local
[DeviceA-security-policy-ip-6-pinglocalout] destination-zone untrust
[DeviceA-security-policy-ip-6-pinglocalout] service ping
[DeviceA-security-policy-ip-6-pinglocalout] action pass
[DeviceA-security-policy-ip-6-pinglocalout] quit
```

# Configure a rule named **pinglocalin** to allow Device A to respond to the ping packets from Device B.

```
[DeviceA-security-policy-ip] rule name pinglocalin
[DeviceA-security-policy-ip-7-pinglocalin] source-zone untrust
[DeviceA-security-policy-ip-7-pinglocalin] destination-zone local
[DeviceA-security-policy-ip-7-pinglocalin] service ping
[DeviceA-security-policy-ip-7-pinglocalin] action pass
[DeviceA-security-policy-ip-7-pinglocalin] quit
[DeviceA-security-policy-ip] quit
```

5. Specify 10.10.10.10 as the source address for BFD echo packets.

```
[DeviceA] bfd echo-source-ip 10.10.10.10
```

6. Configure track entry 1, and associate it with the BFD session to verify the connectivity between Device A and Device B.

```
[DeviceA] track 1 bfd echo interface gigabitethernet 1/0/1 remote ip 10.2.1.2 local
ip 10.2.1.1
[DeviceA-track-1] quit
```

Configuring Device B

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.2.1.2 255.255.255.0
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

# Configure a main static route to 20.1.1.0/24 with next hop 10.2.1.1 and default priority 60, and associate the static route with track entry 1.

```
[DeviceB] ip route-static 20.1.1.0 24 10.2.1.1 track 1
```

# Configure a backup static route to 20.1.1.0/24 with next hop 10.4.1.3 and priority 80.

```
[DeviceB] ip route-static 20.1.1.0 24 10.4.1.3 preference 80
```

3. Add interfaces to security zones.

```
[DeviceB] security-zone name untrust
```

```
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Untrust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Untrust] quit
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceB-security-zone-Trust] quit
```

4. Configure a security policy:

# Configure a rule named **trust-untrust** to permit packets from network 30.1.1.0/24 to network 20.1.1.0/24.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name trust-untrust
[DeviceB-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceB-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-1-trust-untrust] source-ip-subnet 30.1.1.0 24
[DeviceB-security-policy-ip-1-trust-untrust] destination-ip-subnet 20.1.1.0 24
[DeviceB-security-policy-ip-1-trust-untrust] action pass
[DeviceB-security-policy-ip-1-trust-untrust] quit
```

# Configure a rule named **untrust-trust** to permit packets from network 20.1.1.0/24 to network 30.1.1.0/24.

```
[DeviceB-security-policy-ip] rule name untrust-trust
[DeviceB-security-policy-ip-2-untrust-trust] source-zone untrust
[DeviceB-security-policy-ip-2-untrust-trust] destination-zone trust
[DeviceB-security-policy-ip-2-untrust-trust] source-ip-subnet 20.1.1.0 24
[DeviceB-security-policy-ip-2-untrust-trust] destination-ip-subnet 30.1.1.0 24
[DeviceB-security-policy-ip-2-untrust-trust] action pass
[DeviceB-security-policy-ip-2-untrust-trust] quit
```

# Configure a rule named **bfdlocalout** to allow Device B to send BFD echo packets to Device A.

```
[DeviceB-security-policy-ip] rule name bfdlocalout
[DeviceB-security-policy-ip-3-bfdlocalout] source-zone local
[DeviceB-security-policy-ip-3-bfdlocalout] destination-zone untrust
[DeviceB-security-policy-ip-3-bfdlocalout] service bfd-echo
[DeviceB-security-policy-ip-3-bfdlocalout] action pass
[DeviceB-security-policy-ip-3-bfdlocalout] quit
```

# Configure a rule named **bfdlocalin** to allow Device B to receive BFD echo packets.

```
[DeviceB-security-policy-ip] rule name bfdlocalin
[DeviceB-security-policy-ip-4-bfdlocalin] source-zone untrust
[DeviceB-security-policy-ip-4-bfdlocalin] destination-zone local
[DeviceB-security-policy-ip-4-bfdlocalin] service bfd-echo
[DeviceB-security-policy-ip-4-bfdlocalin] action pass
[DeviceB-security-policy-ip-4-bfdlocalin] quit
```

# Configure a rule named **untrust-untrust** to allow Device B to respond to the BFD echo packets from Device A.

```
[DeviceB-security-policy-ip] rule name untrust-untrust
[DeviceB-security-policy-ip-5-untrust-untrust] source-zone untrust
[DeviceB-security-policy-ip-5-untrust-untrust] destination-zone untrust
[DeviceB-security-policy-ip-5-untrust-untrust] source-ip-host 10.10.10.10
[DeviceB-security-policy-ip-5-untrust-untrust] destination-ip-host 10.2.1.1
[DeviceB-security-policy-ip-5-untrust-untrust] action pass
```

```
[DeviceB-security-policy-ip-5-untrust-untrust] quit
```

# Configure a rule named **pinglocalout** to allow Device B to send ping packets to Device A.

```
[DeviceB-security-policy-ip] rule name pinglocalout
[DeviceB-security-policy-ip-6-pinglocalout] source-zone local
[DeviceB-security-policy-ip-6-pinglocalout] destination-zone untrust
[DeviceB-security-policy-ip-6-pinglocalout] service ping
[DeviceB-security-policy-ip-6-pinglocalout] action pass
[DeviceB-security-policy-ip-6-pinglocalout] quit
```

# Configure a rule named **pinglocalin** to allow Device B to respond to the ping packets from Device A.

```
[DeviceB-security-policy-ip] rule name pinglocalin
[DeviceB-security-policy-ip-7-pinglocalin] source-zone untrust
[DeviceB-security-policy-ip-7-pinglocalin] destination-zone local
[DeviceB-security-policy-ip-7-pinglocalin] service ping
[DeviceB-security-policy-ip-7-pinglocalin] action pass
[DeviceB-security-policy-ip-7-pinglocalin] quit
[DeviceB-security-policy-ip] quit
```

5. Specify 1.1.1.1 as the source address of BFD echo packets.

```
[DeviceB] bfd echo-source-ip 1.1.1.1
```

6. Configure track entry 1, and associate it with the BFD session to verify the connectivity between Device B and Device A.

```
[DeviceB] track 1 bfd echo interface gigabitethernet 1/0/1 remote ip 10.2.1.1 local
ip 10.2.1.2
[DeviceB-track-1] quit
```

Configuring Device C

1. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] ip address 10.3.1.3 255.255.255.0
[DeviceC-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

2. Configure settings for routing:

# Configure a static route to 30.1.1.0/24 with next hop 10.4.1.2.

```
[DeviceC] ip route-static 30.1.1.0 24 10.4.1.2
```

# Configure a static route to 20.1.1.0/24 with next hop 10.3.1.1.

```
[DeviceC] ip route-static 20.1.1.0 24 10.3.1.1
```

Verifying the configuration

# Display information about the track entry on Device A.

```
[DeviceA] display track all
Track ID: 1
  State: Positive
  Duration: 0 days 0 hours 0 minutes 32 seconds
  Tracked object type: BFD
  Notification delay: Positive 0, Negative 0 (in seconds)
  Tracked object:
    BFD session mode: Echo
    Outgoing interface: GigabitEthernet1/0/1
    VPN instance name: --
    Remote IP: 10.2.1.2
    Local IP: 10.2.1.1
```

The output shows that the status of the track entry is Positive, indicating that next hop 10.2.1.2 is reachable.

# Display the routing table of Device A.

```
[DeviceA] display ip routing-table

Destinations : 9        Routes : 9

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.2.1.0/24         Direct 0    0            10.2.1.1        GE1/0/1
10.2.1.1/32         Direct 0    0            127.0.0.1       InLoop0
10.3.1.0/24         Direct 0    0            10.3.1.1        GE1/0/2
10.3.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         Direct 0    0            20.1.1.1        GE1/0/3
20.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
30.1.1.0/24         Static 60   0            10.2.1.2        GE1/0/1
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

The output shows that Device A forwards packets to 30.1.1.0/24 through Device B. The master static route has taken effect.

# Remove the IP address of GigabitEthernet 1/0/1 on Device B.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] undo ip address
```

# Display information about the track entry on Device A.

```
[DeviceA] display track all
Track ID: 1
  State: Negative
  Duration: 0 days 0 hours 0 minutes 32 seconds
  Tracked object type: BFD
  Notification delay: Positive 0, Negative 0 (in seconds)
  Tracked object:
    BFD session mode: Echo
    Outgoing interface: GigabitEthernet1/0/1
    VPN instance name: --
    Remote IP: 10.2.1.2
    Local IP: 10.2.1.1
```

The output shows that the status of the track entry is Negative, indicating that next hop 10.2.1.2 is unreachable.

# Display the routing table of Device A.

```
[DeviceA] display ip routing-table

Destinations : 9        Routes : 9
```

```
Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.2.1.0/24         Direct 0    0            10.2.1.1        GE1/0/1
10.2.1.1/32         Direct 0    0            127.0.0.1       InLoop0
10.3.1.0/24         Direct 0    0            10.3.1.1        GE1/0/2
10.3.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         Direct 0    0            20.1.1.1        GE1/0/3
20.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
30.1.1.0/24         Static 80   0            10.3.1.3        GE1/0/2
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

The output shows that Device A forwards packets to 30.1.1.0/24 through Device C. The backup static route has taken effect.

# Verify that the hosts in 20.1.1.0/24 can communicate with the hosts in 30.1.1.0/24 when the master route fails.

```
[DeviceA] ping -a 20.1.1.1 30.1.1.1
Ping 30.1.1.1: 56  data bytes, press CTRL_C to break
Reply from 30.1.1.1: bytes=56 Sequence=1 ttl=254 time=2 ms
Reply from 30.1.1.1: bytes=56 Sequence=2 ttl=254 time=1 ms
Reply from 30.1.1.1: bytes=56 Sequence=3 ttl=254 time=1 ms
Reply from 30.1.1.1: bytes=56 Sequence=4 ttl=254 time=2 ms
Reply from 30.1.1.1: bytes=56 Sequence=5 ttl=254 time=1 ms

--- Ping statistics for 30.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.00% packet loss
round-trip min/avg/max/std-dev = 1/1/2/1 ms
```

# Verify that the hosts in 30.1.1.0/24 can communicate with the hosts in 20.1.1.0/24 when the master route fails.

```
[DeviceB] ping -a 30.1.1.1 20.1.1.1
Ping 20.1.1.1: 56  data bytes, press CTRL_C to break
Reply from 20.1.1.1: bytes=56 Sequence=1 ttl=254 time=2 ms
Reply from 20.1.1.1: bytes=56 Sequence=2 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=3 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=4 ttl=254 time=1 ms
Reply from 20.1.1.1: bytes=56 Sequence=5 ttl=254 time=1 ms

--- Ping statistics for 20.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.00% packet loss
round-trip min/avg/max/std-dev = 1/1/2/1 ms
```

Contents

Configuring BFD

About BFD

Bidirectional forwarding detection (BFD) provides a general-purpose, standard, medium- and protocol-independent fast failure detection mechanism. It can detect and monitor the connectivity of forwarding paths to detect communication failures quickly so that measures can be taken to ensure service continuity and enhance network availability.

BFD can uniformly and quickly detect the failures of the bidirectional forwarding paths between two devices for upper-layer protocols such as routing protocols. The hello mechanism used by upper-layer protocols needs seconds to detect a link failure, while BFD can provide detection measured in milliseconds.

Single-hop detection and multihop detection

BFD can be used for single-hop and multihop detections.

- **Single-hop detection**—Detects the IP connectivity between two directly connected systems.
- **Multihop detection**—Detects any of the paths between two systems. These paths have multiple hops, and might overlap.

BFD session modes

BFD sessions use echo packets and control packets.

Echo packet mode

Echo packets are encapsulated into UDP packets with port number 3785.

The local end of the link sends echo packets to establish BFD sessions and monitor link status. The peer end does not establish BFD sessions and only forwards the packets back to the originating end. If the local end does not receive echo packets from the peer end within the detection time, it considers the session to be down.

In echo packet mode, BFD supports only single-hop detection and the BFD session is independent of the operating mode.

Control packet mode

Control packets are encapsulated into UDP packets with port number 3784 for single-hop detection or port number 4784 for multihop detection.

The two ends of the link negotiate the establishment of BFD sessions by using the session parameters carried in control packets. Session parameters include session discriminators, desired minimum packet sending and receiving intervals, and local BFD session state.

Before a BFD session is established, BFD has two operating modes—active and passive.

- **Active mode**—BFD actively sends BFD control packets regardless of whether any BFD control packet is received from the peer.
- **Passive mode**—BFD does not send control packets until a BFD control packet is received from the peer.

At least one end must operate in active mode for a BFD session to be established.

After a BFD session is established, the two ends can operate in the following BFD operating modes:

- **Asynchronous mode**—The device periodically sends BFD control packets. The device considers that the session is down if it does not receive any BFD control packets within a specific interval.
- **Demand mode**—The device periodically sends BFD control packets with the D bit set. If the peer end is operating in Asynchronous mode (default), the peer end stops sending BFD control packets after receiving control packets with the D bit set. In this case, BFD detects only the connectivity from the local end to the peer end. If the peer end does not receive control packets within the detection time, the session is declared down. If the peer end is operating in Demand mode, both ends stop sending BFD control packets. The system uses other mechanisms such as Hello mechanism and hardware detection to detect links. The Demand mode can be used to reduce the overhead when a large number of BFD sessions exist.

Supported features

Features	Reference
Static routing IS-IS OSPF RIP BGP IP fast reroute (FRR)	*Layer 3—IP Routing Configuration Guide*
IPv6 static routing OSPFv3	*Layer 3—IP Routing Configuration Guide*
PIM	*IP Multicast Configuration Guide*
Track	*Network Management and Monitoring Configuration Guide*.

Protocols and standards

- RFC 5880, *Bidirectional Forwarding Detection (BFD)*
- RFC 5881, *Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)*
- RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*
- RFC 5883, *Bidirectional Forwarding Detection (BFD) for Multihop Paths*

Restrictions and guidelines: BFD configuration

- By default, the device runs BFD version 1 and is compatible with BFD version 0. You cannot change the BFD version to 0 through commands. When the peer device runs BFD version 0, the local device automatically switches to BFD version 0.
- BFD cannot detect a link over NAT.
- After a BFD session is established, the two ends negotiate BFD parameters, including minimum sending interval, minimum receiving interval, initialization mode, and packet authentication, by exchanging negotiation packets. They use the negotiated parameters without affecting the session status.

Configuring BFD sessions in echo packet mode

About this task

A static BFD session can be created manually by using the **bfd static** command or created dynamically when an application module collaborates with BFD.

Restrictions and guidelines

If you also configure uRPF on the device, follow these restrictions and guidelines:

- For the collaboration between an application module and a BFD session in echo packet mode, use an ACL for uRPF to permit echo packets from the peer device. Without the ACL configuration, uRPF will drop these echo packets.
- For a static BFD session, make sure the source IPv4/IPv6 address specified in the session can pass the uRPF check. If the source IPv4/IPv6 address cannot pass the check, uRPF will drop the echo packets from the peer device.

For more information about uRPF, see *Security Configuration Guide*.

Creating a static BFD session

About this task

A static BFD session in echo packet mode can be used to perform single-hop detection and multihop detection.

Restrictions and guidelines

You need to create a static BFD session in echo packet mode on only the local device to perform detection.

When creating a static BFD session, you must specify a peer IPv4 or IPv6 address. The system checks only the format of the IP address but not its correctness. If the peer IPv4 or IPv6 address is incorrect, the static BFD session cannot be established.

Different static BFD sessions cannot have the same local discriminator.

As a best practice, specify the source IP address for echo packets when creating a static BFD session. If you do not specify the source IP address, the device uses the IP address specified in the **bfd echo-source-ip** or **bfd echo-source-ipv6** command as the source IP address of echo packets.

Make sure the source IP address is not on the same network segment as any local interfaces. This avoids the following situations:

- A large number of ICMP redirect packets might be sent from the peer, resulting in link congestion.
- With malformed packet attack detection and prevention enabled, the local end might filter echo packets sent from the peer as malformed packets, resulting in BFD session establishment failure. For more information about malformed packet attack detection and prevention, see attack detection and prevention in *Security Configuration Guide*.

Creating a static BFD session for single-hop detection

1. Enter system view.
   ```
   system-view
   ```
2. Configure the source IP address of echo packets.
   - Configure the source IP address of echo packets.

bfd echo-source-ip *ip-address*

By default, no source IPv4 address is configured for echo packets.

- Configure the source IPv6 address of echo packets.

  bfd echo-source-ipv6 *ipv6-address*

  By default, no source IPv6 address is configured for echo packets.

  The source IPv6 address of echo packets can only be a global unicast address.

3. Create a static BFD session and enter static BFD session view.

   IPv4:

   bfd static *session-name* [**peer-ip** *ipv4-address* **interface** *interface-type interface-number* **destination-ip** *ipv4-address* [**source-ip** *ipv4-address*] **one-arm-echo discriminator** { **local** *local-value* | **auto** }]

   IPv6:

   bfd static *session-name* [**peer-ipv6** *ipv6-address* **interface** *interface-type interface-number* **destination-ipv6** *ipv6-address* [**source-ipv6** *ipv6-address*] **one-arm-echo discriminator** { **local** *local-value* | **auto** }]

Creating a static BFD session for multihop detection

1. Enter system view.

   system-view

2. Configure the source IP address of echo packets.

   - Configure the source IP address of echo packets.

     bfd echo-source-ip *ip-address*

     By default, no source IPv4 address is configured for echo packets.

   - Configure the source IPv6 address of echo packets.

     bfd echo-source-ipv6 *ipv6-address*

     By default, no source IPv6 address is configured for echo packets.

     The source IPv6 address of echo packets can only be a global unicast address.

3. Create a static BFD session and enter static BFD session view.

   IPv4:

   bfd static *session-name* [**peer-ip** *ipv4-address* [**vpn-instance** *vpn-instance-name*] **destination-ip** *ipv4-address* [**source-ip** *ipv4-address*] **one-arm-echo discriminator** { **local** *local-value* | **auto** }]

   IPv6:

   bfd static *session-name* [**peer-ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*] **destination-ipv6** *ipv6-address* [**source-ipv6** *ipv6-address*] **one-arm-echo discriminator** { **local** *local-value* | **auto** }]

Configuring BFD session parameters for single-hop detection

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Set the minimum interval for receiving BFD echo packets.

   bfd min-echo-receive-interval *interval*

   The default setting is 400 milliseconds.

4. Set the detection time multiplier.

   bfd detect-multiplier *value*

   The default setting is 5.

Configuring BFD session parameters for multihop detection

1. Enter system view.

   system-view

2. Set the minimum interval for receiving BFD echo packets.

   bfd multi-hop min-echo-receive-interval *interval*

   The default setting is 400 milliseconds.

3. Set the detection time multiplier.

   bfd multi-hop detect-multiplier *value*

   The default setting is 5.

Configuring BFD sessions in control packet mode

About BFD session creation methods

BFD sessions in control packet mode can be created statically or established dynamically.

BFD sessions are distinguished by the local discriminator and remote discriminator in control packets. The main difference between a statically created session and a dynamically established session is that they obtain the local discriminator and remote discriminator in different ways.

- The local discriminator and remote discriminator of a static BFD session are specified manually in the **bfd static** command or in features associated with BFD.
- The local discriminator of a dynamic BFD session is assigned by the device, and the remote discriminator is obtained during BFD session negotiation. A created session without manually specified local and remote discriminators is a dynamic BFD session.

Restrictions and guidelines

BFD version 0 does not support the following commands:

- **bfd session init-mode**.
- **bfd authentication-mode**.
- **bfd demand enable**.
- **bfd echo enable**.

Configuring a static BFD session

About this task

A static BFD session can be used for single-hop detection and multihop detection.

Restrictions and guidelines for static BFD session configuration

If a static BFD session is created on the remote end, the static BFD session must be created on the local end.

When creating a static BFD session, you must specify a peer IPv4 or IPv6 address. The system checks only the format of the IP address but not its correctness. If the peer IPv4 or IPv6 address is incorrect, the static BFD session cannot be established.

Different static BFD sessions cannot have the same local discriminator.

Creating a static BFD session for single-hop detection of network layer connectivity

1. Enter system view.

   system-view

2. Create a static BFD session and enter static BFD session view.

   IPv4:

   bfd static *session-name* **peer-ip** *ipv4-address* **interface** *interface-type interface-number* **source-ip** *ipv4-address* **discriminator local** *local-value* **remote** *remote-value*

   For a static BFD session to be established, specify the IPv4 address of the peer interface where the static BFD session resides for the **peer-ip** *ipv4-address* option. Specify the IPv4 address of the local interface where the static BFD session resides for the **source-ip** *ipv4-address* option.

   IPv6:

   bfd static *session-name* **peer-ipv6** *ipv6-address* **interface** *interface-type interface-number* **source-ipv6** *ipv6-address* **discriminator local** *local-value* **remote** *remote-value*

   For a static BFD session to be established, specify the IPv6 address of the peer interface where the static BFD session resides for the **peer-ipv6** *ipv6-address* option. Specify the IPv6 address of the local interface where the static BFD session resides for the **source-ipv6** *ipv6-address* option.

Creating a static BFD session for single-hop detection of data link layer connectivity

1. Enter system view.

   system-view

2. Create a static BFD session and enter static BFD session view.

   bfd static *session-name* **peer-ip default-ip interface** *interface-type interface-number* **source-ip** *ip-address* **discriminator local** *discr-value* **remote** *discr-value*

   For a static BFD session to be established, specify the IPv6 address of the local interface where the static BFD session resides for the **source-ip** *ip-address* option.

3. (Optional.) Associate the interface state with the static BFD session.

   process-interface-status

   By default, the state of a static BFD session does not affect the state of the data link layer of the interface.

   Support for this command depends on the device model. For more information, see the command reference.

4. (Optional.) Configure the timer that delays reporting the first static BFD session establishment failure to the data link layer.

   first-fail-timer *seconds*

   By default, the first static BFD session establishment failure is not reported to the data link layer.

This command takes effect only after you configure the **process-interface-status** command.

Support for this command depends on the device model. For more information, see the command reference.

5. (Optional.) Enable special processing for the static BFD session.

   special-processing [**admin-down** | **authentication-change** | **session-up**] *

   By default, all types of special processing are disabled for a static BFD session.

   Support for this command depends on the device model. For more information, see the command reference.

Creating a static BFD session for multihop detection

1. Enter system view.

   system-view

2. Create a static BFD session and enter static BFD session view.

   IPv4:

   bfd static *session-name* **peer-ip** *ipv4-address* [**vpn-instance** *vpn-instance-name*] **source-ip** *ipv4-address* **discriminator local** *local-value* **remote** *remote-value*

   For a static BFD session to be established, specify the IPv4 address of the peer interface where the static BFD session resides for the **peer-ip** *ipv4-address* option. Specify the IPv4 address of the local interface where the static BFD session resides for the **source-ip** *ipv4-address* option.

   IPv6:

   bfd static *session-name* **peer-ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*] **source-ipv6** *ipv6-address* **discriminator local** *local-value* **remote** *remote-value*

   For a static BFD session to be established, specify the IPv6 address of the peer interface where the static BFD session resides for the **peer-ipv6** *ipv6-address* option. Specify the IPv6 address of the local interface where the static BFD session resides for the **source-ipv6** *ipv6-address* option.

Configuring BFD session parameters for single-hop detection

1. Enter system view.

   system-view

2. Specify the mode for establishing a BFD session.

   bfd session init-mode { **active** | **passive** }

   By default, **active** is specified.

3. Enter interface view or static BFD session view.

   - Enter interface view.

     interface *interface-type interface-number*

   - Enter static BFD session view.

     bfd static *session-name*

     The static BFD session must already exist.

4. (Optional.) Configure the authentication mode for single-hop control packets.

bfd authentication-mode { **hmac-md5** | **hmac-mmd5** | **hmac-msha1** | **hmac-sha1** | **m-md5** | **m-sha1** | **md5** | **sha1** | **simple** } *key-id* { **cipher** *cipher-string* | **plain** *plain-string* }

By default, single-hop BFD packets are not authenticated.

5. Enable the Demand BFD session mode.

   bfd demand enable

   By default, the BFD session is in Asynchronous mode.

6. Set the minimum interval for transmitting single-hop BFD control packets.

   bfd min-transmit-interval *interval*

   The default setting is 400 milliseconds.

7. Set the minimum interval for receiving single-hop BFD control packets.

   bfd min-receive-interval *interval*

   The default setting is 400 milliseconds.

8. Set the single-hop detection time multiplier.

   bfd detect-multiplier *value*

   The default setting is 5.

9. (Optional.) Set the delay timer for BFD to notify upper-layer protocols of session establishment failures.

   bfd init-fail-timer *seconds*

   By default, BFD does not notify upper-layer protocols of session establishment failures.

△ **CAUTION:**

For session establishment failures caused by configuration mismatches at the two ends, this command can cause the upper-layer protocol to act incorrectly. Therefore, use this command with caution. BFD status mismatch and BFD authentication configuration mismatch are examples of configuration mismatches.

Configuring BFD session parameters for multihop detection

1. Enter system view.

   system-view

2. Specify the mode for establishing a BFD session.

   bfd session init-mode { **active** | **passive** }

   By default, **active** is specified.

3. Configure the authentication mode for multihop BFD control packets.

   bfd multi-hop authentication-mode { **m-md5** | **m-sha1** | **md5** | **sha1** | **simple** } *key-id* { **cipher** *cipher-string* | **plain** *plain-string* }

   By default, no authentication is performed.

4. Configure the destination port number for multihop BFD control packets.

   bfd multi-hop destination-port *port-number*

   The default setting is 4784.

5. Set the multihop detection time multiplier.

   bfd multi-hop detect-multiplier *value*

   The default setting is 5.

6. Set the minimum interval for receiving multihop BFD control packets.

bfd multi-hop min-receive-interval *interval*

The default setting is 400 milliseconds.

7. Set the minimum interval for transmitting multihop BFD control packets.

   bfd multi-hop min-transmit-interval *interval*

   The default setting is 400 milliseconds.

8. (Optional.) Set the delay timer for BFD to notify upper-layer protocols of session establishment failures.

   bfd init-fail-timer *seconds*

   By default, BFD does not notify upper-layer protocols of session establishment failures.

△ **CAUTION:**

For session establishment failures caused by configuration mismatches at the two ends, this command can cause the upper-layer protocol to act incorrectly. Therefore, use this command with caution. BFD status mismatch and BFD authentication configuration mismatch are examples of configuration mismatches.

Enabling the echo function

About this task

This function enables the local system to periodically send echo packets to the remote system. The remote system loops back the echo packets to the local system without processing them. If the local system does not receive the looped-back echo packets, it declares the BFD session down.

This function is supported only for single-hop detection.

Restrictions and guidelines

This function does not take effect on BFD sessions associated with interface states.

Procedure

1. Enter system view.

   system-view

2. Enter interface view or static BFD session view.

   - Enter interface view.

     interface *interface-type interface-number*

   - Enter static BFD session view.

     bfd static *session-name*

     The static BFD session must be an existing session used to detect data link layer connectivity.

3. Enable the echo function.

   bfd echo [**receive** | **send**] **enable**

   By default, the echo function is disabled.

Associating the interface state with BFD

About this task

By creating a BFD session for single-hop detection through exchange of BFD control packets, this feature implements fast link detection. When BFD detects a link fault, it sets the link layer protocol state to DOWN(BFD). This behavior helps applications relying on the link layer protocol state achieve fast convergence. The source IP address of control packets is specified manually, and the destination IP address is fixed at 224.0.0.184. As a best practice, specify the IP address of the interface as the source IP address. If the interface does not have an IP address, specify a unicast IP address other than 0.0.0.0 as the source IP address.

You can associate the state of the following interfaces with BFD:

- Layer 3 Ethernet interfaces and subinterfaces. For BFD detection to take effect, do not configure this feature on both a Layer 3 Ethernet interface and its subinterface.
- Layer 3 aggregate interfaces, Layer 3 aggregate subinterfaces, and member ports (Layer 3 Ethernet interfaces only) in a Layer 3 aggregation group. For BFD detection to take effect, do not configure this feature on any two of the interface types at the same time.
- VLAN interfaces.

Restrictions and guidelines

The echo function does not take effect on BFD sessions associated with interface states.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Associate the interface state with BFD.

   bfd detect-interface source-ip *ip-address* [**discriminator local** *local-value* **remote** *remote-value*] [**template** *template-name*]

   By default, the interface state is not associated with BFD. BFD does not set the link layer protocol of the interface to DOWN(BFD) state when detecting a failure.

4. (Optional.) Configure the timer that delays reporting the first BFD session establishment failure to the data link layer.

   bfd detect-interface first-fail-timer *seconds*

   By default, the first BFD session establishment failure is not reported to the data link layer.

5. (Optional.) Enable special processing for BFD sessions.

   bfd detect-interface special-processing [**admin-down** | **authentication-change** | **session-up**] *

   By default, all types of special processing are disabled for BFD sessions.

Configuring a BFD template

About this task

Perform this task to specify BFD parameters in a template for sessions without next hops. You can configure BFD parameters for LSPs and PWs through a BFD template.

Procedure

1. Enter system view.

   system-view

2. Create a BFD template and enter BFD template view.

   bfd template *template-name*

3. (Optional.) Configure the authentication mode for BFD control packets.

   bfd authentication-mode { **hmac-md5** | **hmac-mmd5** | **hmac-msha1** | **hmac-sha1** | **m-md5** | **m-sha1** | **md5** | **sha1** | **simple** } *key-id* { **cipher** *cipher-string* | **plain** *plain-string* }

By default, no authentication is performed.

4. Set the detection time multiplier.

   `bfd detect-multiplier` *`value`*

   The default setting is 5.

5. Set the minimum interval for receiving BFD echo packets.

   `bfd min-echo-receive-interval` *`interval`*

   The default setting is 400 milliseconds.

6. Set the minimum interval for receiving BFD control packets.

   `bfd min-receive-interval` *`interval`*

   The default setting is 400 milliseconds.

7. Set the minimum interval for transmitting BFD control packets.

   `bfd min-transmit-interval` *`interval`*

   The default setting is 400 milliseconds.

Enabling SNMP notifications for BFD

About this task

To report critical BFD events to an NMS, enable SNMP notifications for BFD. For BFD event notifications to be sent correctly, you must also configure SNMP as described in *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   `system-view`

2. Enable SNMP notifications for BFD.

   `snmp-agent trap enable bfd`

   By default, SNMP notifications are enabled for BFD.

Display and maintenance commands for BFD

Execute the **`display`** command in any view and the **`reset`** command in user view.

Task	Command
Display BFD session information.	**display bfd session** [**discriminator local** *local-value* | **static name** *session-name* | **verbose**] **display bfd session** [[**dynamic**] [**control** | **echo**] [**ip**] [**state** { **down** | **admin-down** | **init** | **up** }] [**discriminator remote** *remote-value*] [**peer-ip** *ipv4-address* [**vpn-instance** *vpn-instance-name*]] [**verbose**]] **display bfd session** [[**dynamic**] [**control** | **echo**] [**ipv6**] [**state** { **down** | **admin-down** | **init** | **up** }] [**discriminator remote** *remote-value*] [**peer-ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**verbose**]] **display bfd session** [[**dynamic**] [**control** | **echo**] [**state** { **down** | **admin-down** | **init** | **up** }] [**discriminator remote** *remote-value*] [[**peer-ip** *ipv4-address* [**vpn-instance** *vpn-instance-name*]] | [**peer-ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*]]] [**verbose**]] **display bfd session** [[**static**] [**ip**] [**state** { **down** | **admin-down** | **init** | **up** }] [**discriminator remote** *remote-value*] [**peer-ip** *ipv4-address* [**vpn-instance** *vpn-instance-name*]] [**verbose**] **display bfd session** [[**static**] [**ipv6**] [**state** { **down** | **admin-down** | **init** | **up** }] [**discriminator remote** *remote-value*] [**peer-ipv6** *ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**verbose**]
Clear BFD session statistics.	**reset bfd session statistics**

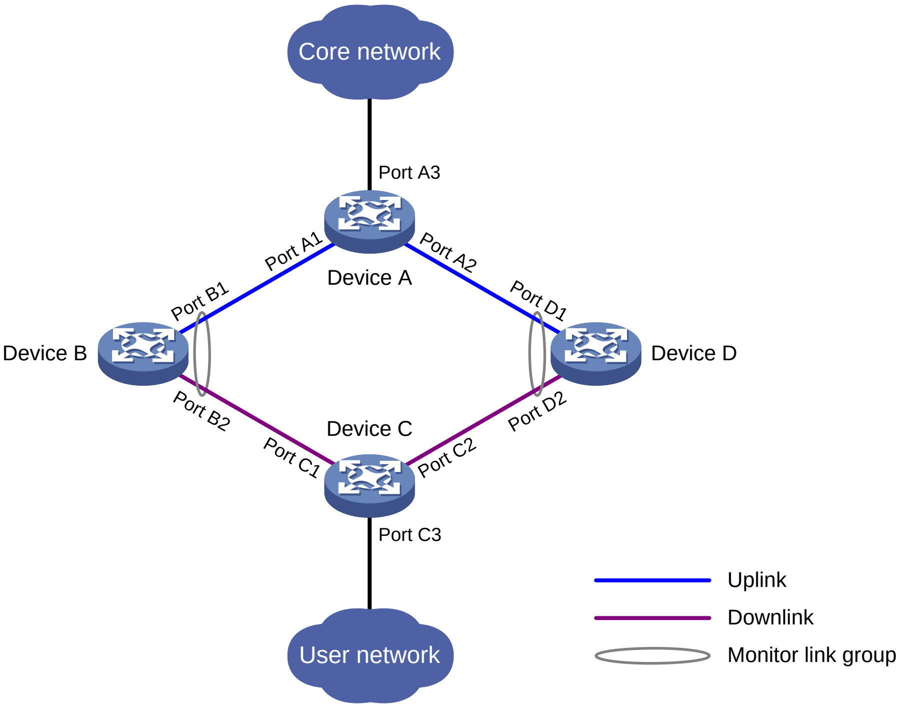
Contents

Configuring Monitor Link

About Monitor Link

Monitor Link is an NSFOCUS-proprietary solution that associates the state of downlink interfaces with the state of uplink interfaces in a monitor link group. When Monitor Link shuts down the downlink interfaces because of an uplink failure, the downstream device changes connectivity to another link.

Figure 1 Monitor Link application scenario



A monitor link group contains uplink and downlink interfaces. An interface can belong to only one monitor link group.

- Uplink interfaces are the monitored interfaces.
- Downlink interfaces are the monitoring interfaces.

As shown in Figure 1:

- Port B1 and Port B2 of Device B form a monitor link group. Port B1 is an uplink interface, and Port B2 is a downlink interface.
- Port D1 and Port D2 of Device D form another monitor link group. Port D1 is an uplink interface, and Port D2 is a downlink interface.

A monitor link group works independently of other monitor link groups. When a monitor link group does not contain any uplink interface or all its uplink interfaces are down, the monitor link group goes down. It forces all downlink interfaces down at the same time. When any uplink interface comes up, the monitor link group comes up and brings up all the downlink interfaces.

Restrictions and guidelines: Monitor Link configuration

Follow these restrictions and guidelines when you configure Monitor Link:

- Do not manually shut down or bring up the downlink interfaces in a monitor link group.
- To avoid frequent state changes of downlink interfaces in the event that uplink interfaces in the monitor link group flap, you can configure a switchover delay. The switchover delay is the time that the downlink interfaces wait before they come up following an uplink interface.

Monitor Link tasks at a glance

To configure Monitor Link, perform the following tasks:

- Enabling Monitor Link globally
- Creating a monitor link group
- Configuring monitor link group member interfaces
- (Optional.) Configuring the switchover delay for the downlink interfaces in a monitor link group

Enabling Monitor Link globally

About this task

All monitor link groups can operate only after you enable Monitor Link globally. When you disable Monitor Link globally, all monitor link groups cannot operate and the downlink interfaces brought down by the monitor link groups resume their original states.

Procedure

1. Enter system view.
   `system-view`
2. Enable Monitor Link globally.
   `undo monitor-link disable`
   By default, Monitor Link is enabled globally.

Creating a monitor link group

1. Enter system view.
   `system-view`
2. Create a monitor link group and enter monitor link group view.
   `monitor-link group` *`group-id`*

Configuring monitor link group member interfaces

Restrictions and guidelines

- An interface can be assigned to only one monitor link group.

- To avoid undesired down/up state changes on the downlink interfaces, configure uplink interfaces before you configure downlink interfaces.
- If you have configured an interface as the downlink interface of a monitor link group, do not configure its subinterfaces as the uplink interfaces of any monitor link group. Otherwise, the Monitor Link operation might be interrupted.
- The state of subinterfaces is associated with the state of the interface. Do not add the interface and its subinterfaces to the same monitor link group. Otherwise, the monitor link group performance might be affected.
- If you have configured a Selected port of an aggregation group as the downlink interface of a monitor link group, do not configure an Unselected port of the aggregation group as the uplink interface of the monitor link group.
- Do not add an aggregate interface and its member ports to the same monitor link group.
- You can configure member interfaces for a monitor link group in monitor link group view or interface view. Configurations made in these views have the same effect. The configuration is supported by the following interfaces:
  - Layer 2 Ethernet interfaces and Layer 2 aggregate interfaces.
  - Layer 3 Ethernet interfaces/subinterfaces.
  - Layer 3 aggregate interfaces/subinterfaces.

Configuring monitor link group member interfaces in monitor link group view

1. Enter system view.

   system-view

2. Enter monitor link group view.

   monitor-link group *group-id*

3. Configure member interfaces for the monitor link group.

   port *interface-type* { *interface-number* | *interface-number.subnumber* } { **downlink** | **uplink** }

   By default, no member interfaces exist in a monitor link group.

Configuring monitor link group member interfaces in interface view

1. Enter system view.

   system-view

2. Enter interface view or subinterface view.

   interface *interface-type* { *interface-number* | *interface-number.subnumber* }

3. Configure the interface as a member of a monitor link group.

   port monitor-link group *group-id* { **downlink** | **uplink** }

   By default, the interface is not a monitor link group member.

Configuring the switchover delay for the downlink interfaces in a monitor link group

1. Enter system view.

   system-view

2. Enter monitor link group view.

   monitor-link group *group-id*

3. Configure the switchover delay for the downlink interfaces in the monitor link group.

   downlink up-delay *delay*

   By default, the switchover delay is 0 seconds. The downlink interfaces come up as soon as an uplink interface comes up.

Display and maintenance commands for Monitor Link

Execute the **display** command in any view:

Task	Command
Display monitor link group information.	**display monitor-link group** { *group-id* | **all** }

Contents

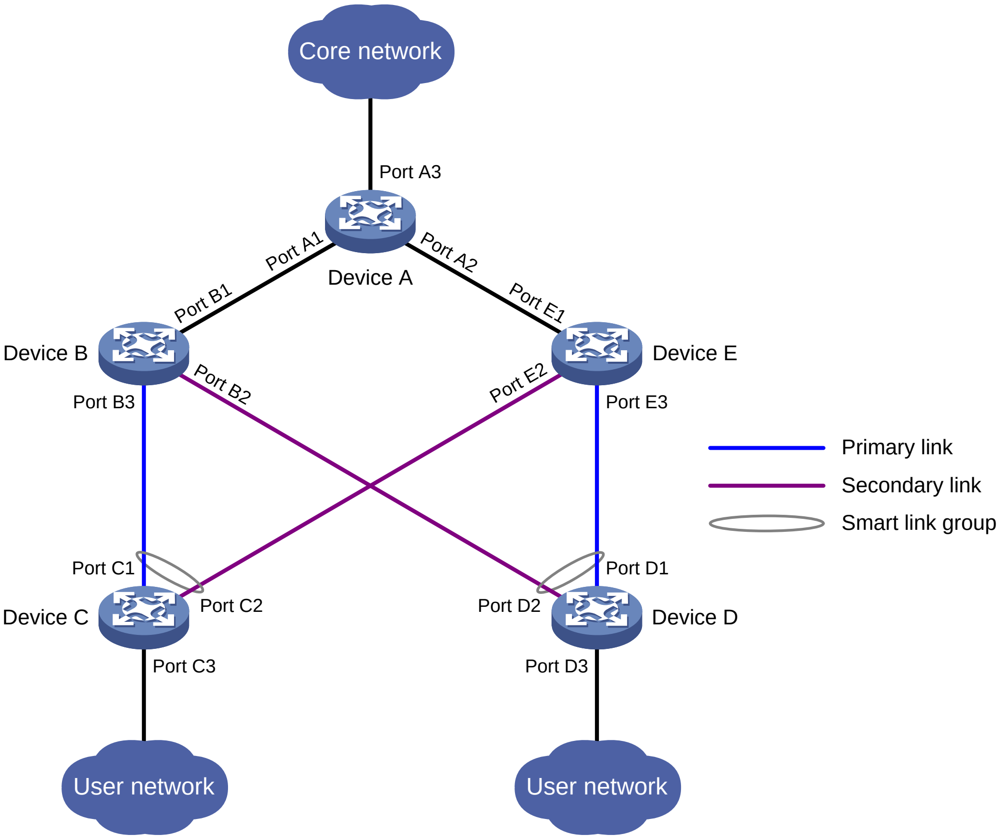
Configuring Smart Link

About Smart Link

Application scenario

Smart Link is an NSFOCUS-proprietary protocol that provides link redundancy and subsecond convergence time in a dual uplink network. As shown in Figure 1, Smart Link is configured on Device C and Device D. The secondary link takes over quickly when the primary link fails.

Figure 1 Dual uplink network diagram



A Smart Link network has the following devices:

- **Smart Link devices**—A Smart Link device has two uplinks. A Smart Link device must be configured with a smart link group and a transmit control VLAN to transmit flush messages. Device C and Device D in Figure 1 are Smart Link devices.
- **Associated devices**—An associated device is an uplink device to which Smart Link devices are connected. An associated device supports Smart Link, and receives flush messages sent from the specified control VLAN. When a primary/secondary link switchover occurs, the associated device updates the MAC address entries and ARP/ND entries according to received flush messages. Device A, Device B, and Device E in Figure 1 are associated devices.

Terminology

Smart link group

A smart link group consists of only two member ports: the primary and the secondary ports. Only one port is active for forwarding at a time, and the other port is blocked and in standby state. When link failure occurs on the active portdue to port shutdown, the standby port becomes active and takes over. The original active port transits to the blocked state.

As shown in Figure 1, Port C1 and Port C2 of Device C form a smart link group. Port C1 is active, and Port C2 is standby. Port D1 and Port D2 of Device D form another smart link group. Port D1 is active, and Port D2 is standby.

Primary/secondary port

Primary port and secondary port are two port types in a smart link group. When both ports in a smart link group are up, the primary port preferentially transits to the forwarding state. The secondary port stays in standby state. When the primary port fails, the secondary port takes over to forward traffic.

As shown in Figure 1, Port C1 of Device C and Port D1 of Device D are primary ports. Port C2 of Device C and Port D2 of Device D are secondary ports.

Primary/secondary link

The link that connects the primary port in a smart link group is the primary link. The link that connects the secondary port is the secondary link.

Flush message

When link switchover occurs, the smart link group uses flush messages to notify other devices to refresh their MAC address entries and ARP/ND entries. Flush messages are common multicast data packets, and will be dropped by a blocked receiving port.

Protected VLAN

A smart link group controls the forwarding state of protected VLANs. Each smart link group on a port controls a different protected VLAN. The state of the port in a protected VLAN is determined by the state of the port in the smart link group.

Transmit control VLAN

The transmit control VLAN is used for transmitting flush messages. When link switchover occurs, the devices (such as Device C and Device D in Figure 1) send flush messages within the transmit control VLAN.

Receive control VLAN

The receive control VLAN is used for receiving and processing flush messages. When link switchover occurs, the devices (such as Device A, Device B, and Device E in Figure 1) receive and process flush messages in the receive control VLAN. In addition, they refresh their MAC address entries and ARP/ND entries.

How Smart Link works

Link backup

As shown in Figure 1, the link on Port C1 of Device C is the primary link. The link on Port C2 of Device C is the secondary link. Port C1 is in forwarding state, and Port C2 is in standby state. When the primary link fails, Port C2 takes over to forward traffic and Port C1 is blocked and placed in standby state.

When a port switches to the forwarding state, the system outputs log information to notify the user of the port state change.

Topology change

Link switchover might outdate the MAC address entries and ARP/ND entries on all devices. A flush update mechanism is provided to ensure correct packet transmission. With this mechanism, a Smart Link-enabled device updates its information by transmitting flush messages over the backup link to its upstream devices. This mechanism requires the upstream devices to be capable of recognizing Smart Link flush messages to update their MAC address forwarding entries and ARP/ND entries.

Preemption mode

As shown in Figure 1, the link on Port C1 of Device C is the primary link. The link on Port C2 of Device C is the secondary link. When the primary link fails, Port C1 is automatically blocked and placed in standby state, and Port C2 takes over to forward traffic. When the primary link recovers, one of the following actions occurs:

- If the smart link group is not configured with a preemption mode, Port C1 stays blocked to keep traffic forwarding stable. Port C1 does not take over to forward traffic until the next link switchover.
- If the smart link group is configured with a preemption mode and the preemption conditions are met, Port C1 takes over to forward traffic as soon as its link recovers. Port C2 is automatically blocked and placed in standby state.

Load sharing

A ring network might carry traffic of multiple VLANs. Smart Link can forward traffic from different VLANs in different smart link groups for load sharing.

To implement load sharing, you can assign a port to multiple smart link groups. Configure each group with a different protected VLAN. Make sure the state of the port is different in these smart link groups, so traffic from different VLANs can be forwarded along different paths.

You can configure protected VLANs for a smart link group by referencing Multiple Spanning Tree Instances (MSTIs). For more information about MSTIs, see *Layer 2—LAN Switching Configuration Guide*.

Collaboration between Smart Link and Monitor Link for port status detection

Smart Link cannot detect when faults occur on the uplink of the upstream devices or when faults are cleared. You can configure the Monitor Link function to monitor the status of the uplink ports of the upstream devices. Monitor Link adapts the up/down state of downlink ports to uplink ports, and triggers Smart Link to perform link switchover on the downstream device. For more information about Monitor Link, see *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines: Smart Link configuration

If you configure a port as both an aggregation group member and a smart link group member, only the aggregation group configuration takes effect. The port is not shown in the output from the **`display smart-link group`** command. The smart link group configuration takes effect after the port leaves the aggregation group.

Smart Link tasks at a glance

To configure Smart Link, perform the following tasks:

1. Configuring a Smart Link device

   a. Configuring protected VLANs for a smart link group
   b. Configuring member ports for a smart link group
   c. (Optional.) Configuring a preemption mode for a smart link group
   d. (Optional.) Enabling the sending of flush messages
2. Enabling an associated device to receive flush messages

Configuring a Smart Link device

Prerequisites for Smart Link device configuration

Before configuring a Smart Link device, complete the following tasks:

- To prevent loops, shut down a port before configuring it as a smart link group member. You can bring up the port only after completing the smart link group configuration.
- Disable the spanning tree feature on the ports you want to add to the smart link group.

Configuring protected VLANs for a smart link group

Prerequisites

Before you configure protected VLANs, you must configure an MST region and the VLAN-to-instance mapping table. For more information about MST regions, see spanning tree configuration in *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.
   system-view
2. Create a smart link group and enter smart link group view.
   smart-link group *group-id*
3. Configure protected VLANs for the smart link group.
   protected-vlan reference-instance *instance-id-list*

Configuring member ports for a smart link group

Restrictions and guidelines

You can configure member ports for a smart link group either in smart link group view or in interface view. The configurations made in these two views have the same effect.

In smart link group view

1. Enter system view.
   system-view
2. Create a smart link group and enter smart link group view.
   smart-link group *group-id*
3. Configure member ports for a smart link group.
   port *interface-type interface-number* { **primary** | **secondary** }
   By default, no member port is configured for a smart link group.

In interface view

1. Enter system view.

system-view

2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.

   interface *interface-type interface-number*

3. Configure member ports for a smart link group.

   port smart-link group *group-id* { **primary** | **secondary** }

   By default, an interface is not a smart link group member.

Configuring a preemption mode for a smart link group

1. Enter system view.

   system-view

2. Enter smart link group view.

   smart-link group *group-id*

3. Configure a preemption mode for the smart link group.

   preemption mode { **role** | **speed** [**threshold** *threshold-value*] }

   By default, preemption is disabled.

4. Configure the preemption delay.

   preemption delay *delay*

   By default, the preemption delay is 1 second.

   The preemption delay configuration takes effect only after a preemption mode is configured.

Enabling the sending of flush messages

Restrictions and guidelines

- The control VLAN configured for a smart link group must be different from the control VLAN configured for any other smart link group.
- Make sure the configured control VLAN already exists, and assign the smart link group member ports to the control VLAN.
- The control VLAN of a smart link group must also be one of its protected VLANs. Do not remove the control VLAN. Otherwise, flush messages cannot be sent correctly.

Procedure

1. Enter system view.

   system-view

2. Enter smart link group view.

   smart-link group *group-id*

3. Enable flush update.

   flush enable [**control-vlan** *vlan-id*]

   By default, flush update is enabled, and VLAN 1 is the control VLAN.

Enabling an associated device to receive flush messages

Restrictions and guidelines

- You do not need to enable all ports on the associated devices to receive flush messages. Enable the feature only on all control VLANs of ports on the primary and secondary links between the Smart Link device and the destination device.
- If no control VLAN is specified for processing flush messages, the device forwards the received flush messages without any processing.
- Make sure the receive control VLAN is the same as the transmit control VLAN configured on the Smart Link device. If they are not the same, the associated device will forward the received flush messages directly without any processing.
- Do not remove the control VLANs. Otherwise, flush messages cannot be sent correctly.
- Make sure the control VLANs are existing VLANs, and assign the ports capable of receiving flush messages to the control VLANs.

Prerequisites

Disable the spanning tree feature on the associated device's ports that connect to the member ports of the smart link group. Otherwise, the ports will discard flush messages when they are not in forwarding state if a topology change occurs.

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.

   `interface` *`interface-type interface-number`*

3. Configure the control VLANs for receiving flush messages.

   `smart-link flush enable` [**`control-vlan`** *`vlan-id-list`*]

   By default, no control VLAN receives flush messages.

Display and maintenance commands for Smart Link

Execute **`display`** commands in any view and the **`reset`** command in user view:

Task	Command
Display information about the received flush messages.	**`display smart-link flush`**
Display smart link group information.	**`display smart-link group`** { *`group-id`* | **`all`** }
Clear the statistics about flush messages.	**`reset smart-link statistics`**

Contents

Configuring interface backup

About interface backup

Interface backup enables you to configure multiple backup interfaces for a Layer 3 interface to increase link availability. When the primary interface fails or is overloaded, its backup interfaces can take over or participate in traffic forwarding.

Compatible interfaces

Table 1 Interfaces that support interface backup

Category	Interfaces	Remarks
Ethernet	Layer 3 Ethernet interfaces/subinterfaces	N/A
Others	Dialer interfaces Tunnel interfaces	A dialer interface can be used as the primary interface only when it is a PPPoE client in permanent session mode.

Backup modes

The primary interface and its backup interfaces can operate in strict active/standby mode or load sharing mode.

- **Strict active/standby mode**—Only one interface transmits traffic. All the other interfaces are in STANDBY state.
- **Load sharing mode**—Backup interfaces participate in traffic forwarding when the amount of traffic on the primary interface reaches the upper threshold. They are activated and deactivated depending on the amount of traffic.

In strict active/standby mode, traffic loss occurs when the active interface is overloaded. Load sharing mode improves link efficiency and reduces the risk of packet loss.

Strict active/standby mode

In strict active/standby mode, the primary interface always has higher priority than all backup interfaces.

- When the primary interface is operating correctly, all traffic is transmitted through the primary interface.
- When the primary interface fails, the highest-priority backup interface takes over. If the highest-priority backup interface also fails, the second highest-priority backup interface takes over, and so forth.

NOTE:

If two backup interfaces have the same priority, the one configured first has preference.

An active backup interface is always preempted by the primary interface. However, a higher-priority backup interface cannot preempt a lower-priority backup interface that has taken over the primary interface.

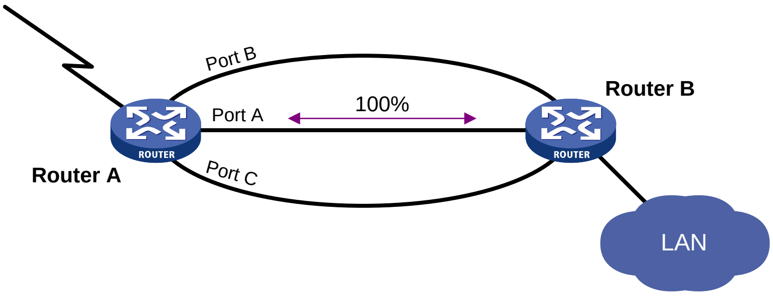
- The primary interface takes over when it recovers from a failure condition.

- The higher-priority backup interface cannot take over when it recovers from a failure condition while the primary interface is still down.

As shown in Figure 1, Port A on Router A is the primary interface. Port B (with a priority of 30) and Port C (with a priority of 20) are its backup interfaces.

- When Port A is operating correctly, all traffic is transmitted through Port A.
- When Port A fails, Port B takes over because it has higher priority than Port C. If Port B also fails, Port C takes over.
- When Port A is recovered, it preempts the active backup interface because it is the primary interface. If Port B is recovered while Port A is still down, Port B cannot preempt Port C to forward traffic.

Figure 1 Strict active/backup mode



Load sharing mode

In load sharing mode, the backup interfaces are activated to transmit traffic depending on the traffic load on the primary interface.

- When the amount of traffic on the primary interface exceeds the upper threshold, the backup interfaces are activated in descending order of priority. This action continues until the traffic drops below the upper threshold.
- When the total amount of traffic on all load-shared interfaces decreases below the lower threshold, the backup interfaces are deactivated in ascending order of priority. This action continues until the total amount of traffic exceeds the lower threshold.
- When the primary interface fails (in DOWN state), the strict active/standby mode applies. Only one backup interface can forward traffic.

The upper and lower thresholds are user configurable.

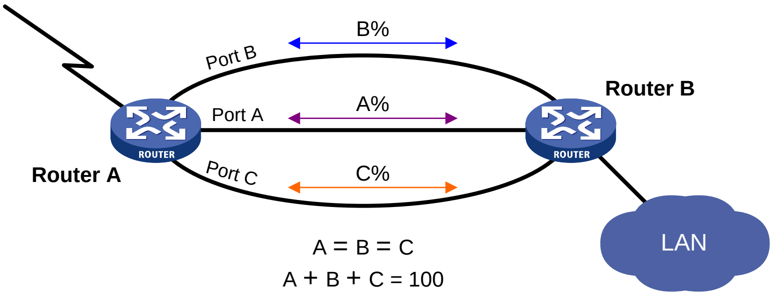
NOTE:

- "Traffic" on an interface refers to the amount of incoming or outgoing traffic, whichever is higher.
- If two backup interfaces have the same priority, the one configured first has preference.

As shown in Figure 2, Port A on Router A is the primary interface. Port B (with a priority of 30) and Port C (with a priority of 20) are its backup interfaces.

- When the amount of traffic on Port A exceeds the upper threshold, Port B is activated, because it has higher priority than Port C. If the amount of traffic on Port A still exceeds the upper threshold, Port C is activated.
- When the total amount of traffic on all load-shared interfaces decreases below the lower threshold, Port C is first deactivated, because its priority is lower than Port B. If the total amount of traffic on Port A and Port B is still below the lower threshold, Port B is deactivated.

Figure 2 Load sharing mode



Restrictions and guidelines: Interface backup configuration

When you configure interface backup, follow these restrictions and guidelines:

- The device supports up to 10 primary interfaces.
- An interface can be configured as a backup only for one interface.
- An interface cannot be both a primary and backup interface.
- The strict active/standby mode and load sharing mode cannot be configured at the same time.

Interface backup tasks at a glance

To configure interface backup, perform the following tasks:

- Configuring strict active/standby interface backup

  Choose one of the following tasks:

  - Explicitly specifying backup interfaces without traffic thresholds

    Use this method if you want to monitor the interface state of the primary interface for a switchover to occur.

  - Using interface backup with the Track module

    Use this method if you want to monitor any other state, such as the link state of the primary interface.

- Configuring load-shared interface backup

Prerequisites for configuring interface backup

Make sure the primary and backup interfaces have routes to the destination network.

Explicitly specifying backup interfaces without traffic thresholds

About this task

Perform this task if you want to monitor the interface state of the primary interface for a switchover to occur. For the primary and backup interfaces to operate in strict active/standby mode, do not specify the traffic thresholds on the primary interface. If the traffic thresholds are configured, the interfaces will operate in load sharing mode.

You can assign priority to backup interfaces. When the primary interface fails, the backup interfaces are activated in descending order of priority, with the highest-priority interface activated first. If two backup interfaces have the same priority, the one configured first has preference.

To prevent link flapping from causing frequent interface switchovers, you can configure the following switchover delay timers:

- **Up delay timer**—Number of seconds that the primary or backup interface must wait before it can come up.
- **Down delay timer**—Number of seconds that the active primary or backup interface must wait before it is set to down state.

When the link of the active interface fails, the interface state does not change immediately. Instead, a down delay timer starts. If the link recovers before the timer expires, the interface state does not change. If the link is still down when the timer expires, the interface state changes to down.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

   This interface must be the primary interface.

3. Specify a backup interface.

   ```
   backup interface interface-type interface-number [ priority ]
   ```

   By default, an interface does not have any backup interfaces.

   Repeat this command to specify up to three backup interfaces for the interface.

4. Set the switchover delay timers.

   ```
   backup timer delay up-delay down-delay
   ```

   By default, the up and down delay timers are both 5 seconds.

Using interface backup with the Track module

About this task

Perform this task if you want to monitor any other state, such as the link state of the primary interface. To use interface backup with the Track module to provide strict active/standby backup for a primary interface:

- Configure a track entry to monitor state information of the primary interface. For example, monitor its link state.
- Associate the track entry with a backup interface.

Interface backup changes the state of the backup interface in response to the track entry state, as shown in Table 2.

Table 2 Action on the backup interface in response to the track entry state change

Track entry state	State of the monitored primary link	Action on the backup interface
Positive	The primary link is operating correctly.	Places the backup interface in STANDBY state.
Negative	The primary link has failed.	Activates the backup interface to take over.

Track entry state	State of the monitored primary link	Action on the backup interface
NotReady	The primary link is not monitored. This situation occurs when the track module or the monitoring module is not ready, for example, because the Track module is restarting or the monitoring settings are incomplete. In this situation, interface backup cannot obtain information about the primary link from the track module.	• If the track entry state stays in NotReady state after it is created, interface backup does not change the state of the backup interface. • If the track entry state changes to NotReady from Positive or Negative, the backup interface changes back to the forwarding state before it was used for interface backup.

For more information about configuring a track entry, see Track configuration in *Network Management and Monitoring Configuration Guide*.

Restrictions and guidelines

- You can associate an interface with only one track entry.
- You can create the associated track entry before or after the association. The association takes effect after the track entry is created.
- To maintain performance, limit the number of associations to 64.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

   This interface must be the interface you are using as a backup.

3. Associate the interface with a track entry.

   `backup track` *`track-entry-number`*

   By default, an interface is not associated with a track entry.

Configuring load-shared interface backup

About this task

To implement load-balanced interface backup, you must configure the traffic thresholds on the primary interface. Interface backup regularly compares the amount of traffic with the thresholds to determine whether to activate or deactivate a backup interface. The traffic polling interval is user configurable.

You can assign priority to backup interfaces.

- When the amount of traffic on the primary interface exceeds the upper threshold, the backup interfaces are activated in descending order of priority.
- When the total amount of traffic on all load-shared interfaces decreases below the lower threshold, the backup interfaces are deactivated in ascending order of priority.

If two backup interfaces have the same priority, the one configured first has preference.

If a traffic flow has a fast forwarding entry, all packets of the flow will be forwarded out of the outgoing interface in the entry. The packets of the flow will not be distributed between interfaces when the upper threshold is reached. For more information about fast forwarding, see *Layer 3—IP Services Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

   You must enter the view of the primary interface.

3. Configure a backup interface for the interface.

   backup interface *interface-type interface-number* [*priority*]

   By default, an interface does not have any backup interfaces.

   Repeat this command to specify up to three backup interfaces.

4. Set backup load sharing thresholds.

   backup threshold *upper-threshold lower-threshold*

   By default, no traffic thresholds are configured.

5. Set the traffic polling interval.

   backup timer flow-check *interval*

   The default interval is 30 seconds.

Display and maintenance commands for interface backup

Execute **display** commands in any view.

Task	Command
Display traffic statistics for load-shared interfaces.	**display interface-backup statistics**
Display the status of primary and backup interfaces.	**display interface-backup state**

Contents

Configuring interface collaboration

About interface collaboration

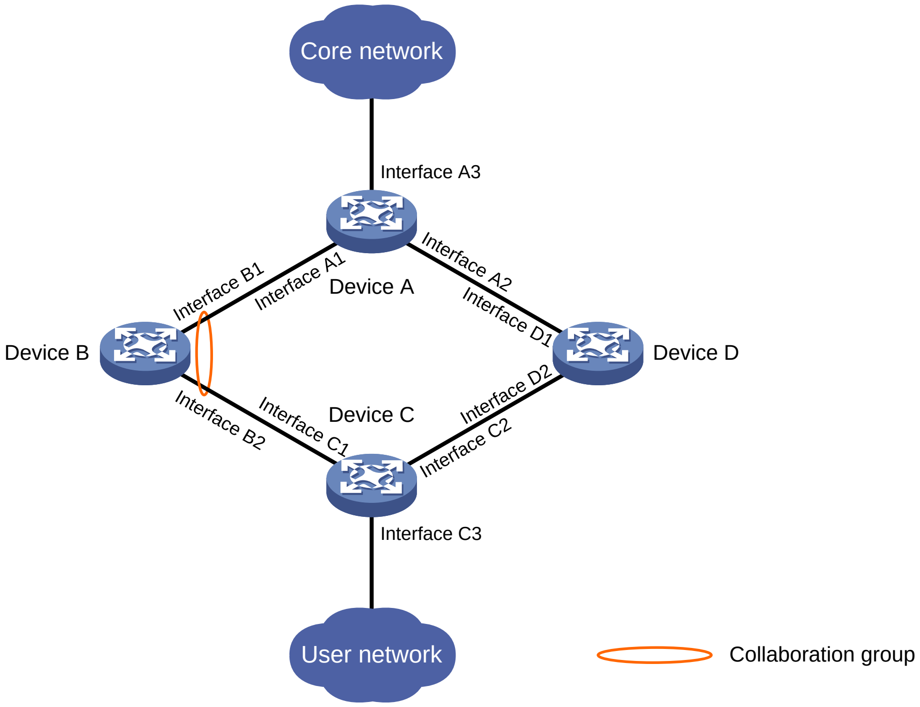
The interface collaboration feature assigns different interfaces on a device to a collaboration group and associates the states of these interfaces. All member interfaces in a collaboration group can or cannot transmit packets at the same time. This feature is typically used to associate the states of downlink and uplink interfaces.

Typical application

As shown in Figure 1, Device C accesses the core network through Device B. When Interface B1 goes down, the switchover of traffic from Device C to Device D is slow because Interface B2 is still up.

If Interface B1 and Interface B2 belong to one collaboration group, Device B brings down Interface B2 when Interface B1 goes down to achieve fast traffic switchover. Similarly, the device brings down Interface B1 when Interface B2 goes down.

Figure 1 Interface collaboration application scenario



How interface collaboration works

The interface collaboration feature works as follows:

- When any member interface in a collaboration group goes down, the device sets all other member interfaces in the collaboration group to the **Collaboration-down** state. The state of the collaboration group is down, and no member interfaces in the collaboration group can transmit packets.

- When any member interface in **DOWN** or **Collaboration-down** state comes up, the device attempts to bring up all other member interfaces in the collaboration group.
  - If all other member interfaces come up in 10 seconds, the collaboration group comes up. All member interfaces in the collaboration group can transmit packets.
  - If any member interface cannot come up in 10 seconds, the device sets that member interface to **DOWN** state and sets all other member interfaces to the **Collaboration-down** state. The collaboration group goes down, and no member interfaces in the collaboration group can transmit packets.

Restrictions and guidelines: Interface collaboration configuration

This feature cannot be used together with the external security service bypass feature. For more information about external security service bypass see Layer 2 forwarding configuration in *Layer 2—LAN Switching Configuration Guide*.

Interface collaboration tasks at a glance

To configure interface collaboration, perform the following tasks:

- Creating a collaboration group
- Configuring collaboration group member interfaces
- (Optional.) Configuring the delay for the member interfaces in a collaboration group
- (Optional.) Removing ineffective member interfaces from all collaboration groups

Prerequisites for interface collaboration

Before configuring interface collaboration, you must enable Monitor Link globally (see *Network Monitoring and Management Configuration Guide*).

Creating a collaboration group

1. Enter system view.
   `system-view`
2. Create a collaboration group and enter collaboration group view.
   `collaboration-group` *`group-id`*
   By default, no collaboration groups exist.

Configuring collaboration group member interfaces

Restrictions and guidelines

An interface can belong to only one collaboration group.

For a collaboration group to work correctly, do not assign its member interfaces to a redundancy group.

Do not add both an aggregate interface and its member ports to the same collaboration group.

Do not add both Selected ports and Unselected ports in a dynamic aggregation group to the same collaboration group.

If you have configured a member interface as the uplink/downlink interface of a monitor link group, do not configure any other member interface in the same collaboration group as the downlink/uplink interface of any monitor link group.

You can assign only one interface of a link to a collaboration group.

You can configure member interfaces for a collaboration group in collaboration group view or interface view. Configurations made in these views have the same effect.

Configuring a collaboration group member interface in collaboration group view

1. Enter system view.

   system-view

2. Enter collaboration group view.

   collaboration-group *group-id*

3. Configure an interface as a member of the collaboration group.

   port *interface-type interface-number*

   By default, no member interfaces exist in a collaboration group.

Configuring a collaboration group member interface in interface view

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the interface as a member of a collaboration group.

   port collaboration-group *group-id*

   By default, an interface is not a collaboration group member.

Configuring the delay for the member interfaces in a collaboration group to come up

About this task

By default, member interfaces in a collaboration group come up and forward service traffic immediately after the device restarts. Traffic loss occurs if the service modules have not returned to normal state.

Perform this task to enable the member interfaces to come up after a configurable delay time upon a device restart.

Procedure

1. Enter system view.

   system-view

2. Enter collaboration group view.

   collaboration-group *group-id*

3. Configure the delay for the member interfaces in a collaboration group to come up.

   up-delay *delay*

   By default, the delay time is 0 seconds. The member interfaces come up as soon as the device restarts.

Removing ineffective member interfaces from all collaboration groups

About this task

A member interface in a collaboration group becomes ineffective when the card that hosts the interface is removed or changed to another slot or the ID of the IRF member device that hosts the interface changes. This task prevents an ineffective interface from causing all other member interfaces in the same collaboration group to go down.

An ineffective interface cannot be automatically assigned to the original collaboration group when its hosting card is reinstalled or changed back to the original slot or the IRF member ID is changed back to the original ID. You must assign it to the original collaboration group manually.

Procedure

1. Enter system view.

   system-view

2. Enter collaboration group view.

   collaboration-group clean

Display and maintenance commands for interface collaboration

Execute the **display** command in any view:

Task	Command
Display collaboration group information.	**display collaboration-group** { *group-id* | **all** } [**verbose**]

Contents

Using ping, tracert, and system debugging

This chapter covers ping, tracert, and information about debugging the system.

Ping

About ping

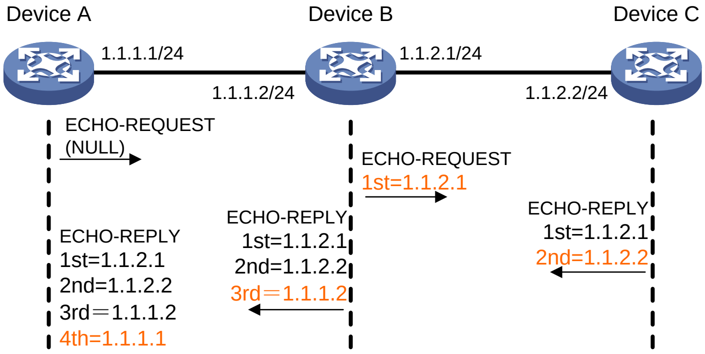
Use the ping utility to determine if an address is reachable.

Ping sends ICMP echo requests (ECHO-REQUEST) to the destination device. Upon receiving the requests, the destination device responds with ICMP echo replies (ECHO-REPLY) to the source device. The source device outputs statistics about the ping operation, including the number of packets sent, number of echo replies received, and the round-trip time. You can measure the network performance by analyzing these statistics.

You can use the **`ping –r`** command to display the routers through which ICMP echo requests have passed. The test procedure of **`ping –r`** is as shown in Figure 1:

1. The source device (Device A) sends an ICMP echo request to the destination device (Device C) with the RR option empty.
2. The intermediate device (Device B) adds the IP address of its outbound interface (1.1.2.1) to the RR option of the ICMP echo request, and forwards the packet.
3. Upon receiving the request, the destination device copies the RR option in the request and adds the IP address of its outbound interface (1.1.2.2) to the RR option. Then the destination device sends an ICMP echo reply.
4. The intermediate device adds the IP address of its outbound interface (1.1.1.2) to the RR option in the ICMP echo reply, and then forwards the reply.
5. Upon receiving the reply, the source device adds the IP address of its inbound interface (1.1.1.1) to the RR option. The detailed information of routes from Device A to Device C is formatted as: 1.1.1.1 <-> {1.1.1.2; 1.1.2.1} <-> 1.1.2.2.

Figure 1 Ping operation



Using a ping command to test network connectivity

Perform the following tasks in any view:

- Determine if an IPv4 address is reachable.

ping [**ip**] [**-a** *source-ip* | **-c** *count* | **-f** | **-h** *ttl* | **-i** *interface-type interface-number* | **-m** *interval* | **-n** | **-p** *pad* | **-q** | **-r** | **-s** *packet-size* | **-t** *timeout* | **-tos** *tos* | **-v** | **-vpn-instance** *vpn-instance-name*] * *host*

Increase the timeout time (indicated by the **-t** keyword) on a low-speed network.

- Determine if an IPv6 address is reachable.

  ping ipv6 [**-a** *source-ipv6* | **-c** *count* | **-i** *interface-type interface-number* | **-m** *interval* | **-q** | **-s** *packet-size* | **-t** *timeout* | **-tc** *traffic-class* | **-v** | **-vpn-instance** *vpn-instance-name*] * *host*

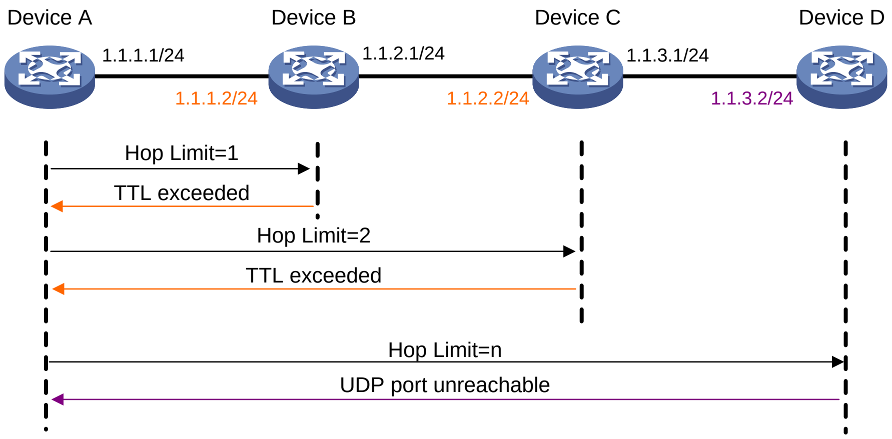
  Increase the timeout time (indicated by the **-t** keyword) on a low-speed network.

Tracert

About tracert

Tracert (also called Traceroute) enables retrieval of the IP addresses of Layer 3 devices in the path to a destination. In the event of network failure, use tracert to test network connectivity and identify failed nodes.

Figure 2 Tracert operation



Tracert uses received ICMP error messages to get the IP addresses of devices. Tracert works as shown in Figure 2:

6. The source device sends a UDP packet with a TTL value of 1 to the destination device. The destination UDP port is not used by any application on the destination device.
7. The first hop (Device B, the first Layer 3 device that receives the packet) responds by sending a TTL-expired ICMP error message to the source, with its IP address (1.1.1.2) encapsulated. This way, the source device can get the address of the first Layer 3 device (1.1.1.2).
8. The source device sends a packet with a TTL value of 2 to the destination device.
9. The second hop (Device C) responds with a TTL-expired ICMP error message, which gives the source device the address of the second Layer 3 device (1.1.2.2).
10. This process continues until a packet sent by the source device reaches the ultimate destination device. Because no application uses the destination port specified in the packet, the destination device responds with a port-unreachable ICMP message to the source device, with its IP address encapsulated. This way, the source device gets the IP address of the destination device (1.1.3.2).
11. The source device determines that:
    - The packet has reached the destination device after receiving the port-unreachable ICMP message.

- The path to the destination device is 1.1.1.2 to 1.1.2.2 to 1.1.3.2.

Prerequisites

Before you use a tracert command, perform the tasks in this section.

For an IPv4 network:

- Enable sending of ICMP timeout packets on the intermediate devices (devices between the source and destination devices). If the intermediate devices are NSFOCUS devices, execute the **ip ttl-expires enable** command on the devices. For more information about this command, see IP performance optimazition commands in *Layer 3—IP Services Command Reference*.
- Enable sending of ICMP destination unreachable packets on the destination device. If the destination device is an NSFOCUS device, execute the **ip unreachables enable** command. For more information about this command, see IP performance optimization commands in *Layer 3—IP Services Command Reference*.

For an IPv6 network:

- Enable sending of ICMPv6 timeout packets on the intermediate devices (devices between the source and destination devices). If the intermediate devices are NSFOCUS devices, execute the **ipv6 hoplimit-expires enable** command on the devices. For more information about this command, see IPv6 basics commands in *Layer 3—IP Services Command Reference*.
- Enable sending of ICMPv6 destination unreachable packets on the destination device. If the destination device is an NSFOCUS device, execute the **ipv6 unreachables enable** command. For more information about this command, see IPv6 basics commands in *Layer 3—IP Services Command Reference*.

Using a tracert command to identify failed or all nodes in a path

Perform the following tasks in any view:

- Trace the route to an IPv4 destination.

  tracert [**-a** *source-ip* | **-f** *first-ttl* | **-i** *interface-type interface-number* | **-m** *max-ttl* | **-p** *port* | **-q** *packet-number* | **-t** *tos* | **-vpn-instance** *vpn-instance-name* [**-resolve-as** { **global** | **none** | **vpn** }] | **-w** *timeout*] * *host*

- Trace the route to an IPv6 destination.

  tracert ipv6 [**-f** *first-hop* | **-i** *interface-type interface-number* | **-m** *max-hops* | **-p** *port* | **-q** *packet-number* | **-t** *traffic-class* | **-vpn-instance** *vpn-instance-name* [**-resolve-as** { **global** | **none** | **vpn** }] | **-w** *timeout*] * *host*

System debugging

About system debugging

The device supports debugging for the majority of protocols and features, and provides debugging information to help users diagnose errors.

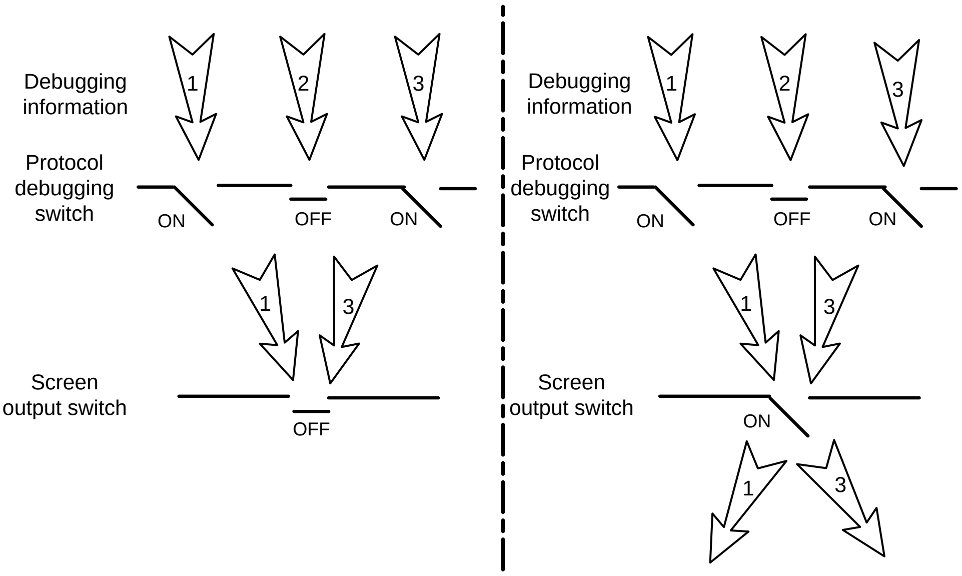
The following switches control the display of debugging information:

- **Module debugging switch**—Controls whether to generate the module-specific debugging information.
- **Screen output switch**—Controls whether to display the debugging information on a certain screen. Use the **`terminal monitor`** and **`terminal logging level`** commands to turn on the screen output switch. For more information about the commands, see information center commands in *Network Management and Monitoring Command Reference*.

As shown in Figure 3, the device can provide debugging for the three modules 1, 2, and 3. The debugging information can be output on a terminal only when both the module debugging switch and the screen output switch are turned on.

Debugging information is typically displayed on a console. You can also send debugging information to other destinations. For more information, see "Configuring the information center."

Figure 3 Relationship between the module and screen output switch



Configuring debugging

Restrictions and guidelines

△ **CAUTION:**

Output of excessive debugging messages increases the CPU usage and downgrades the system performance. To guarantee system performance, enable debugging only for modules that are in an exceptional condition.

Enable debugging for modules for troubleshooting purposes. When debugging is complete, use the **`undo debugging all`** command to disable all the debugging functions.

Procedure

Perform the following tasks in user view:

1. Enable debugging for a module.

   `debugging` *`module-name`* [*`option`*]

   By default, debugging is disabled for all modules.

2. (Optional.) Enable the debugging-auto-off feature to automatically disable all types of debugging when the CPU usage reaches or exceeds the lowest CPU usage alarm threshold.

   `debugging-auto-off enable cpu-usage-alarm`

By default, the debugging-auto-off feature is disabled.

Display and maintenance commands for debugging

Execute **display** commands in any view.

Task	Command
Display the enabled debugging features.	**display debugging** [*module-name*]
Display the enabling status of the debugging-auto-off feature.	**display debugging-auto-off**

Contents

Configuring NTP

The device is inadequate in clock precision. As a best practice, do not use the device as a time server to synchronize the time of the other devices.

About NTP

NTP is used to synchronize system clocks among distributed time servers and clients on a network. NTP runs over UDP and uses UDP port 123.

NTP application scenarios

Various tasks, including network management, charging, auditing, and distributed computing depend on accurate and synchronized system time setting on the network devices. NTP is typically used in large networks to dynamically synchronize time among network devices.

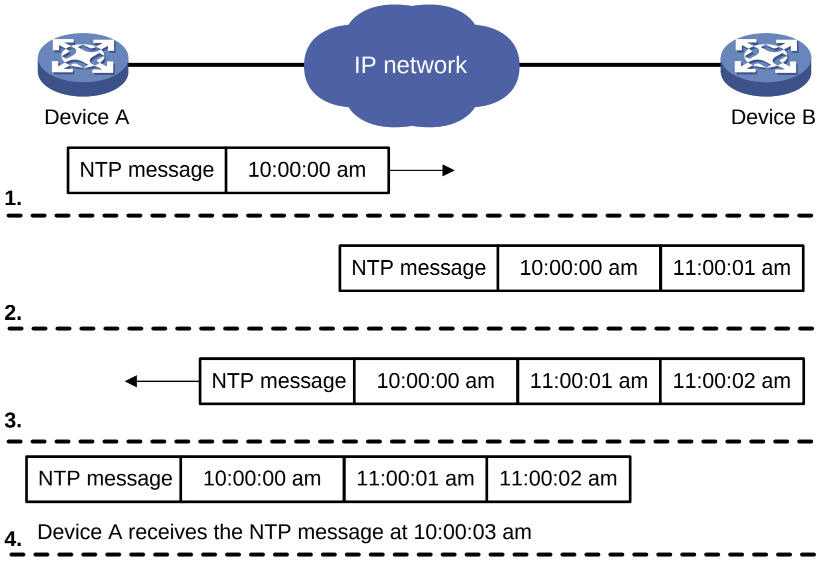
NTP guarantees higher clock accuracy than manual system clock setting. In a small network that does not require high clock accuracy, you can keep time synchronized among devices by changing their system clocks one by one.

NTP working mechanism

Figure 1 shows how NTP synchronizes the system time between two devices (Device A and Device B, in this example). Assume that:

- Prior to the time synchronization, the time is set to 10:00:00 am for Device A and 11:00:00 am for Device B.
- Device B is used as the NTP server. Device A is to be synchronized to Device B.
- It takes 1 second for an NTP message to travel from Device A to Device B, and from Device B to Device A.
- It takes 1 second for Device B to process the NTP message.

Figure 1 Basic work flow



The synchronization process is as follows:

1. Device A sends Device B an NTP message, which is timestamped when it leaves Device A. The time stamp is 10:00:00 am (T1).
2. When this NTP message arrives at Device B, Device B adds a timestamp showing the time when the message arrived at Device B. The timestamp is 11:00:01 am (T2).
3. When the NTP message leaves Device B, Device B adds a timestamp showing the time when the message left Device B. The timestamp is 11:00:02 am (T3).
4. When Device A receives the NTP message, the local time of Device A is 10:00:03 am (T4).

Up to now, Device A can calculate the following parameters based on the timestamps:

- The roundtrip delay of the NTP message: Delay = (T4 – T1) – (T3 – T2) = 2 seconds.
- Time difference between Device A and Device B: Offset = [(T2 – T1) + (T3 – T4)] /2 = 1 hour.

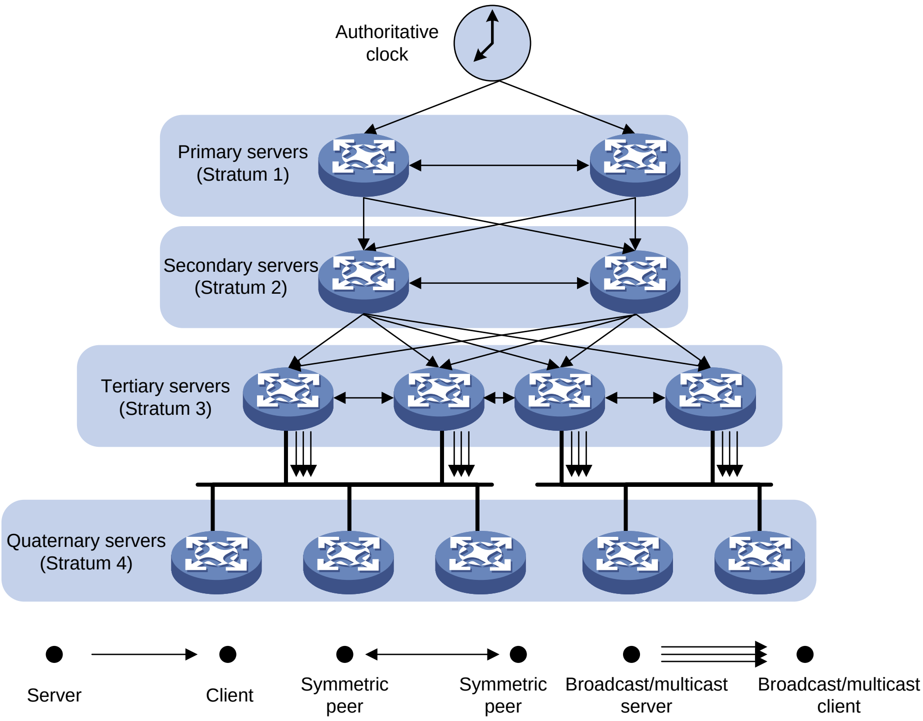
Based on these parameters, Device A can be synchronized to Device B.

This is only a rough description of the work mechanism of NTP. For more information, see the related protocols and standards.

NTP architecture

NTP uses stratums 1 to 16 to define clock accuracy, as shown in Figure 2. A lower stratum value represents higher accuracy. Clocks at stratums 1 through 15 are in synchronized state, and clocks at stratum 16 are not synchronized.

Figure 2 NTP architecture


Authoritative clock
Primary servers (Stratum 1)
Secondary servers (Stratum 2)
Tertiary servers (Stratum 3)
Quaternary servers (Stratum 4)
Server
Client
Symmetric peer
Symmetric peer
Broadcast/multicast server
Broadcast/multicast client

A stratum 1 NTP server gets its time from an authoritative time source, such as an atomic clock. It provides time for other devices as the primary NTP server. A stratum 2 time server receives its time from a stratum 1 time server, and so on.

To ensure time accuracy and availability, you can specify multiple NTP servers for a device. The device selects an optimal NTP server as the clock source based on parameters such as stratum. The clock that the device selects is called the reference source. For more information about clock selection, see the related protocols and standards.

If the devices in a network cannot synchronize to an authoritative time source, you can perform the following tasks:

- Select a device that has a relatively accurate clock from the network.
- Use the local clock of the device as the reference clock to synchronize other devices in the network.

NTP association modes

NTP supports the following association modes:

- Client/server mode
- Symmetric active/passive mode
- Broadcast mode
- Multicast mode

You can select one or more association modes for time synchronization. Table 1 provides detailed description for the four association modes.

In this document, an "NTP server" or a "server" refers to a device that operates as an NTP server in client/server mode. Time servers refer to all the devices that can provide time synchronization, including NTP servers, NTP symmetric peers, broadcast servers, and multicast servers.

Table 1 NTP association modes

Mode	Working process	Principle	Application scenario
Client/server	On the client, specify the IP address of the NTP server. A client sends a clock synchronization message to the NTP servers. Upon receiving the message, the servers automatically operate in server mode and send a reply. If the client can be synchronized to multiple time servers, it selects an optimal clock and synchronizes its local clock to the optimal reference source after receiving the replies from the servers.	A client can synchronize to a server, but a server cannot synchronize to a client.	As Figure 2 shows, this mode is intended for configurations where devices of a higher stratum synchronize to devices with a lower stratum.
Symmetric active/passive	On the symmetric active peer, specify the IP address of the symmetric passive peer. A symmetric active peer periodically sends clock synchronization messages to a symmetric passive peer. The symmetric passive peer automatically operates in symmetric passive mode and sends a reply. If the symmetric active peer can be synchronized to multiple time servers, it selects an optimal clock and synchronizes its local clock to the optimal reference source after receiving the replies from the servers.	A symmetric active peer and a symmetric passive peer can be synchronized to each other. If both of them are synchronized, the peer with a higher stratum is synchronized to the peer with a lower stratum.	As Figure 2 shows, this mode is most often used between servers with the same stratum to operate as a backup for one another. If a server fails to communicate with all the servers of a lower stratum, the server can still synchronize to the servers of the same stratum.

Mode	Working process	Principle	Application scenario
Broadcast	A server periodically sends clock synchronization messages to the broadcast address 255.255.255.255. Clients listen to the broadcast messages from the servers to synchronize to the server according to the broadcast messages. When a client receives the first broadcast message, the client and the server start to exchange messages to calculate the network delay between them. Then, only the broadcast server sends clock synchronization messages.	A broadcast client can synchronize to a broadcast server, but a broadcast server cannot synchronize to a broadcast client.	A broadcast server sends clock synchronization messages to synchronize clients in the same subnet. As Figure 2 shows, broadcast mode is intended for configurations involving one or a few servers and a potentially large client population. The broadcast mode has lower time accuracy than the client/server and symmetric active/passive modes because only the broadcast servers send clock synchronization messages.
Multicast	A multicast server periodically sends clock synchronization messages to the user-configured multicast address. Clients listen to the multicast messages from servers and synchronize to the server according to the received messages.	A multicast client can synchronize to a multicast server, but a multicast server cannot synchronize to a multicast client.	A multicast server can provide time synchronization for clients in the same subnet or in different subnets. The multicast mode has lower time accuracy than the client/server and symmetric active/passive modes.

NTP security

To improve time synchronization security, NTP provides the access control and authentication functions.

NTP access control

You can control NTP access by using an ACL. The access rights are in the following order, from the least restrictive to the most restrictive:

- **Peer**—Allows time requests and NTP control queries (such as alarms, authentication status, and time server information) and allows the local device to synchronize itself to a peer device.
- **Server**—Allows time requests and NTP control queries, but does not allow the local device to synchronize itself to a peer device.
- **Synchronization**—Allows only time requests from a system whose address passes the access list criteria.
- **Query**—Allows only NTP control queries from a peer device to the local device.

When the device receives an NTP request, it matches the request against the access rights in order from the least restrictive to the most restrictive: **peer**, **server**, **synchronization**, and **query**.

- If no NTP access control is configured, the **peer** access right applies.
- If the IP address of the peer device matches a **permit** statement in an ACL, the access right is granted to the peer device. If a **deny** statement or no ACL is matched, no access right is granted.
- If no ACL is specified for an access right or the ACL specified for the access right is not created, the access right is not granted.

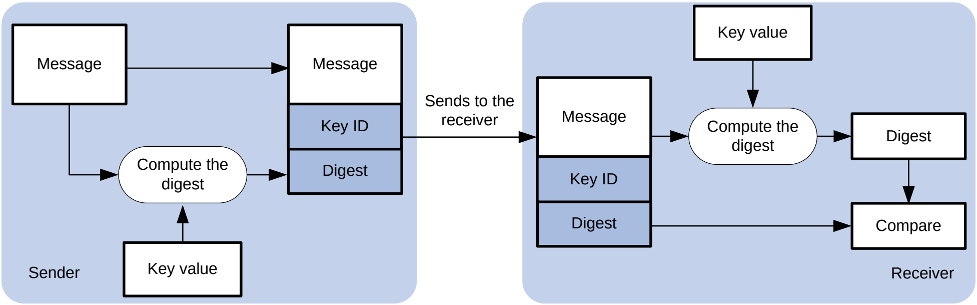
- If none of the ACLs specified for the access rights is created, the **peer** access right applies.
- If none of the ACLs specified for the access rights contains rules, no access right is granted.

This feature provides minimal security for a system running NTP. A more secure method is NTP authentication.

NTP authentication

Use this feature to authenticate the NTP messages for security purposes. If an NTP message passes authentication, the device can receive it and get time synchronization information. If not, the device discards the message. This function makes sure the device does not synchronize to an unauthorized time server.

Figure 3 NTP authentication



As shown in Figure 3, NTP authentication is performed as follows:

1. The sender uses the key identified by the key ID to calculate a digest for the NTP message through the MD5 authentication algorithm. Then it sends the calculated digest together with the NTP message and key ID to the receiver.
2. Upon receiving the message, the receiver performs the following actions:
   a. Finds the key according to the key ID in the message.
   b. Uses the key and the MD5 authentication algorithm to calculate the digest for the message.
   c. Compares the digest with the digest contained in the NTP message.
      – If they are different, the receiver discards the message.
      – If they are the same, the local device determines whether the sender is allowed to use the authentication ID. If the sender is allowed to use the authentication ID, the receiver accepts the message. If the sender is not allowed to use the authentication ID, the receiver discards the message.

Protocols and standards

- RFC 1305, *Network Time Protocol (Version 3) Specification, Implementation and Analysis*
- RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*

Restrictions and guidelines: NTP configuration

- You cannot configure both NTP and SNTP on the same device.
- NTP is supported only on the following Layer 3 interfaces:
  - Layer 3 Ethernet interfaces.
  - Layer 3 Ethernet subinterfaces.
  - Layer 3 aggregate interfaces.
  - Layer 3 aggregate subinterfaces.

  - VLAN interfaces.
  - Tunnel interfaces.
- Do not configure NTP settings on an aggregate member port.
- To avoid frequent time changes or even synchronization failures, do not specify more than one reference source on a network.
- For correct time synchronization, make sure the time offset between the system time and the NTP clock source is less than 68 years.
- You must specify a context in the **clock protocol** command for using NTP to obtain the time. For more information about the **clock protocol** command, see device management commands in *Fundamentals Command Reference*.
- You can configure NTP on only one context.

NTP tasks at a glance

To configure NTP, perform the following tasks:

1. Enabling the NTP service
2. Configuring NTP association mode
   - Configuring NTP in client/server mode
   - Configuring NTP in symmetric active/passive mode
   - Configuring NTP in broadcast mode
   - Configuring NTP in multicast mode
3. (Optional.) Configuring the local clock as the reference source
4. (Optional.) Configuring access control rights
5. (Optional.) Configuring NTP authentication
   - Configuring NTP authentication in client/server mode
   - Configuring NTP authentication in symmetric active/passive mode
   - Configuring NTP authentication in broadcast mode
   - Configuring NTP authentication in multicast mode
6. (Optional.) Controlling NTP packet sending and receiving
   - Specifying the source interface for NTP messages
   - Disabling an interface from receiving NTP messages
   - Configuring the maximum number of dynamic associations
   - Setting a DSCP value for NTP packets
7. (Optional.) Controlling output of logs and traps during time synchronization

Enabling the NTP service

Restrictions and guidelines

NTP and SNTP are mutually exclusive. Before you enable NTP, make sure SNTP is disabled.

Procedure

1. Enter system view.
   ```
   system-view
   ```
2. Enable the NTP service.
   ```
   ntp-service enable
   ```

By default, the NTP service is disabled.

Configuring NTP association mode

Configuring NTP in client/server mode

Restrictions and guidelines

To configure NTP in client/server mode, specify an NTP server for the client.

For a client to synchronize to an NTP server, make sure the server is synchronized by other devices or uses its local clock as the reference source.

If the stratum level of a server is higher than or equal to a client, the client will not synchronize to that server.

You can specify multiple servers for a client by executing the **ntp-service unicast-server** or **ntp-service ipv6 unicast-server** command multiple times.

Procedure

1. Enter system view.

   system-view

2. Specify an NTP server for the device.

   IPv4:

   ntp-service unicast-server { *server-name* | *ip-address* } [**vpn-instance** *vpn-instance-name*] [**authentication-keyid** *keyid* | **maxpoll** *maxpoll-interval* | **minpoll** *minpoll-interval* | **priority** | **source** *interface-type interface-number* | **version** *number*] *

   IPv6:

   ntp-service ipv6 unicast-server { *server-name* | *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**authentication-keyid** *keyid* | **maxpoll** *maxpoll-interval* | **minpoll** *minpoll-interval* | **priority** | **source** *interface-type interface-number*] *

   By default, no NTP server is specified.

Configuring NTP in symmetric active/passive mode

Restrictions and guidelines

To configure NTP in symmetric active/passive mode, specify a symmetric passive peer for the active peer.

For a symmetric passive peer to process NTP messages from a symmetric active peer, execute the **ntp-service enable** command on the symmetric passive peer to enable NTP.

For time synchronization between the symmetric active peer and the symmetric passive peer, make sure either or both of them are in synchronized state.

You can specify multiple symmetric passive peers by executing the **ntp-service unicast-peer** or **ntp-service ipv6 unicast-peer** command multiple times.

Procedure

1. Enter system view.

   system-view

2. Specify a symmetric passive peer for the device.

IPv4:

```
ntp-service unicast-peer { peer-name | ip-address } [ vpn-instance vpn-instance-name ] [ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll minpoll-interval | priority | source interface-type interface-number | version number ] *
```

IPv6:

```
ntp-service ipv6 unicast-peer { peer-name | ipv6-address } [ vpn-instance vpn-instance-name ] [ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll minpoll-interval | priority | source interface-type interface-number ] *
```

By default, no symmetric passive peer is specified.

Configuring NTP in broadcast mode

Restrictions and guidelines

To configure NTP in broadcast mode, you must configure an NTP broadcast client and an NTP broadcast server.

For a broadcast client to synchronize to a broadcast server, make sure the broadcast server is synchronized by other devices or uses its local clock as the reference source.

Configuring the broadcast client

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Configure the device to operate in broadcast client mode.

   ```
   ntp-service broadcast-client
   ```

   By default, the device does not operate in any NTP association mode.

   After you execute the command, the device receives NTP broadcast messages from the specified interface.

Configuring the broadcast server

1. Enter system view.

   ```
   system-view
   ```

2. Enter interface view.

   ```
   interface interface-type interface-number
   ```

3. Configure the device to operate in NTP broadcast server mode.

   ```
   ntp-service broadcast-server [ authentication-keyid keyid | version number ] *
   ```

   By default, the device does not operate in any NTP association mode.

   After you execute the command, the device sends NTP broadcast messages from the specified interface.

Configuring NTP in multicast mode

Restrictions and guidelines

To configure NTP in multicast mode, you must configure an NTP multicast client and an NTP multicast server.

For a multicast client to synchronize to a multicast server, make sure the multicast server is synchronized by other devices or uses its local clock as the reference source.

Configuring a multicast client

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the device to operate in multicast client mode.

   IPv4:

   ntp-service multicast-client [*ip-address*]

   IPv6:

   ntp-service ipv6 multicast-client *ipv6-address*

   By default, the device does not operate in any NTP association mode.

   After you execute the command, the device receives NTP multicast messages from the specified interface.

Configuring the multicast server

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Configure the device to operate in multicast server mode.

   IPv4:

   ntp-service multicast-server [*ip-address*] [**authentication-keyid** *keyid* | **ttl** *ttl-number* | **version** *number*] *

   IPv6:

   ntp-service ipv6 multicast-server *ipv6-address* [**authentication-keyid** *keyid* | **ttl** *ttl-number*] *

   By default, the device does not operate in any NTP association mode.

   After you execute the command, the device sends NTP multicast messages from the specified interface.

Configuring the local clock as the reference source

About this task

This task enables the device to use the local clock as the reference so that the device is synchronized.

Restrictions and guidelines

Make sure the local clock can provide the time accuracy required for the network. After you configure the local clock as the reference source, the local clock is synchronized, and can operate as a time server to synchronize other devices in the network. If the local clock is incorrect, timing errors occur.

The system time reverts to the initial BIOS default after a reboot. As a best practice, do not configure the local clock as the reference source or configure the device as a time server.

The system time reverts to the initial BIOS default after a cold reboot. The system clock stops and does not record the passing of time during a warm reboot. As a best practice, do not configure the local clock as the reference source or configure the device as a time server.

Devices differ in clock precision. As a best practice to avoid network flapping and clock synchronization failure, configure only one reference clock on the same network segment and make sure the clock has high precision.

Prerequisites

Before you configure this feature, adjust the local system time to ensure that it is accurate.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the local clock as the reference source.

   ntp-service refclock-master [*ip-address*] [*stratum*]

   By default, the device does not use the local clock as the reference source.

Configuring access control rights

Prerequisites

Before you configure the right for peer devices to access the NTP services on the local device, create and configure ACLs associated with the access right. For information about configuring an ACL, see *ACL and QoS Configuration Guide*.

Restrictions and guidelines

Follow the restrictions and guidelines as described in Table 2 to configure the NTP access control rights.

Table 2 Restrictions and guidelines for configuring access control rights

Access control right	Whether the time can be synchronized (whether configurable on a client)	Whether to synchronize the time of other devices (whether configurable on a time server)	Whether control queries are allowed
Peer	Yes	Yes	Yes
Server	Yes	No	Yes
Synchronization	Yes	No	No
Query	No	No	Yes

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure the right for peer devices to access the NTP services on the local device.

   IPv4:

   ntp-service access { **peer** | **query** | **server** | **synchronization** } **acl** *ipv4-acl-number*

   IPv6:

```
ntp-service ipv6 { peer |query | server | synchronization } acl
ipv6-acl-number
```

By default, the right for peer devices to access the NTP services on the local device is **peer**.

Configuring NTP authentication

Configuring NTP authentication in client/server mode

Restrictions and guidelines

To ensure a successful NTP authentication in client/server mode, configure the same authentication key ID and key on the server and client. Make sure the peer device is allowed to use the key ID for authentication on the local device.

NTP authentication results differ when different configurations are performed on client and server. For more information, see Table 3. (N/A in the table means that whether the configuration is performed or not does not make any difference.)

Table 3 NTP authentication results

Client			Server	
Enable NTP authentication	**Specify the server and key**	**Trusted key**	**Enable NTP authentication**	**Trusted key**
Successful authentication				
Yes	Yes	Yes	Yes	Yes
Failed authentication				
Yes	Yes	Yes	Yes	No
Yes	Yes	Yes	No	N/A
Yes	Yes	No	N/A	N/A
Authentication not performed				
Yes	No	N/A	N/A	N/A
No	N/A	N/A	N/A	N/A

Configuring NTP authentication for a client

1. Enter system view.

```
system-view
```

2. Enable NTP authentication.

```
ntp-service authentication enable
```

By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

```
ntp-service authentication-keyid keyid authentication-mode md5
{ cipher | simple } string [ acl ipv4-acl-number | ipv6 acl
ipv6-acl-number ] *
```

By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

```
ntp-service reliable authentication-keyid keyid
```

By default, no authentication key is configured as a trusted key.

5. Associate the specified key with an NTP server.

   IPv4:

   ntp-service unicast-server { *server-name* | *ip-address* } [**vpn-instance** *vpn-instance-name*] **authentication-keyid** *keyid*

   IPv6:

   ntp-service ipv6 unicast-server { *server-name* | *ipv6-address* } [**vpn-instance** *vpn-instance-name*] **authentication-keyid** *keyid*

Configuring NTP authentication for a server

1. Enter system view.

   system-view

2. Enable NTP authentication.

   ntp-service authentication enable

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ntp-service authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

   By default, no authentication key is configured as a trusted key.

Configuring NTP authentication in symmetric active/passive mode

Restrictions and guidelines

To ensure a successful NTP authentication in symmetric active/passive mode, configure the same authentication key ID and key on the active peer and passive peer. Make sure the peer device is allowed to use the key ID for authentication on the local device.

NTP authentication results differ when different configurations are performed on active peer and passive peer. For more information, see Table 4. (N/A in the table means that whether the configuration is performed or not does not make any difference.)

Table 4 NTP authentication results

Active peer				Passive peer	
Enable NTP authentication	**Specify the peer and key**	**Trusted key**	**Stratum level**	**Enable NTP authentication**	**Trusted key**
Successful authentication					
Yes	Yes	Yes	N/A	Yes	Yes
Failed authentication					
Yes	Yes	Yes	N/A	Yes	No
Yes	Yes	Yes	N/A	No	N/A
Yes	No	N/A	N/A	Yes	N/A
No	N/A	N/A	N/A	Yes	N/A

Active peer				Passive peer	
Enable NTP authentication	**Specify the peer and key**	**Trusted key**	**Stratum level**	**Enable NTP authentication**	**Trusted key**
Yes	Yes	No	Larger than the passive peer	N/A	N/A
Yes	Yes	No	Smaller than the passive peer	Yes	N/A
Authentication not performed					
Yes	No	N/A	N/A	No	N/A
No	N/A	N/A	N/A	No	N/A
Yes	Yes	No	Smaller than the passive peer	No	N/A

Configuring NTP authentication for an active peer

1. Enter system view.

   ```
   system-view
   ```

2. Enable NTP authentication.

   ```
   ntp-service authentication enable
   ```

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ```
   ntp-service authentication-keyid keyid authentication-mode md5
   { cipher | simple } string [ acl ipv4-acl-number | ipv6 acl
   ipv6-acl-number ] *
   ```

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ```
   ntp-service reliable authentication-keyid keyid
   ```

   By default, no authentication key is configured as a trusted key.

5. Associate the specified key with a passive peer.

   IPv4:

   ```
   ntp-service unicast-peer { ip-address | peer-name } [ vpn-instance
   vpn-instance-name ] authentication-keyid keyid
   ```

   IPv6:

   ```
   ntp-service ipv6 unicast-peer { ipv6-address | peer-name }
   [ vpn-instance vpn-instance-name ] authentication-keyid keyid
   ```

Configuring NTP authentication for a passive peer

1. Enter system view.

   ```
   system-view
   ```

2. Enable NTP authentication.

   ```
   ntp-service authentication enable
   ```

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ```
   ntp-service authentication-keyid keyid authentication-mode md5
   { cipher | simple } string [ acl ipv4-acl-number | ipv6 acl
   ipv6-acl-number ] *
   ```

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

   By default, no authentication key is configured as a trusted key.

Configuring NTP authentication in broadcast mode

Restrictions and guidelines

To ensure a successful NTP authentication in broadcast mode, configure the same authentication key ID and key on the broadcast server and client. Make sure the peer device is allowed to use the key ID for authentication on the local device.

NTP authentication results differ when different configurations are performed on broadcast client and server. For more information, see Table 5. (N/A in the table means that whether the configuration is performed or not does not make any difference.)

Table 5 NTP authentication results

Broadcast server			**Broadcast client**	
Enable NTP authentication	**Specify the server and key**	**Trusted key**	**Enable NTP authentication**	**Trusted key**
Successful authentication				
Yes	Yes	Yes	Yes	Yes
Failed authentication				
Yes	Yes	Yes	Yes	No
Yes	Yes	Yes	No	N/A
Yes	Yes	No	Yes	N/A
Yes	No	N/A	Yes	N/A
No	N/A	N/A	Yes	N/A
Authentication not performed				
Yes	Yes	No	No	N/A
Yes	No	N/A	No	N/A
No	N/A	N/A	No	N/A

Configuring NTP authentication for a broadcast client

1. Enter system view.

   system-view

2. Enable NTP authentication.

   ntp-service authentication enable

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ntp-service authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

By default, no authentication key is configured as a trusted key.

Configuring NTP authentication for a broadcast server

1. Enter system view.

   system-view

2. Enable NTP authentication.

   ntp-service authentication enable

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ntp-service authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

   By default, no authentication key is configured as a trusted key.

5. Enter interface view.

   interface *interface-type interface-number*

6. Associate the specified key with the broadcast server.

   ntp-service broadcast-server authentication-keyid *keyid*

   By default, the broadcast server is not associated with a key.

Configuring NTP authentication in multicast mode

Restrictions and guidelines

To ensure a successful NTP authentication in multicast mode, configure the same authentication key ID and key on the multicast server and client. Make sure the peer device is allowed to use the key ID for authentication on the local device.

NTP authentication results differ when different configurations are performed on broadcast client and server. For more information, see Table 6. (N/A in the table means that whether the configuration is performed or not does not make any difference.)

Table 6 NTP authentication results

Multicast server			Multicast client	
Enable NTP authentication	**Specify the server and key**	**Trusted key**	**Enable NTP authentication**	**Trusted key**
Successful authentication				
Yes	Yes	Yes	Yes	Yes
Failed authentication				
Yes	Yes	Yes	Yes	No
Yes	Yes	Yes	No	N/A
Yes	Yes	No	Yes	N/A
Yes	No	N/A	Yes	N/A
No	N/A	N/A	Yes	N/A

Multicast server			Multicast client	
Enable NTP authentication	**Specify the server and key**	**Trusted key**	**Enable NTP authentication**	**Trusted key**
Authentication not performed				
Yes	Yes	No	No	N/A
Yes	No	N/A	No	N/A
No	N/A	N/A	No	N/A

Configuring NTP authentication for a multicast client

1. Enter system view.

   system-view

2. Enable NTP authentication.

   ntp-service authentication enable

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ntp-service authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

   By default, no authentication key is configured as a trusted key.

Configuring NTP authentication for a multicast server

1. Enter system view.

   system-view

2. Enable NTP authentication.

   ntp-service authentication enable

   By default, NTP authentication is disabled.

3. Configure an NTP authentication key.

   ntp-service authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no NTP authentication key exists.

4. Configure the key as a trusted key.

   ntp-service reliable authentication-keyid *keyid*

   By default, no authentication key is configured as a trusted key.

5. Enter interface view.

   interface *interface-type interface-number*

6. Associate the specified key with a multicast server.

   IPv4:

   ntp-service multicast-server [*ip-address*] **authentication-keyid** *keyid*

   IPv6:

   ntp-service ipv6 multicast-server *ipv6-multicast-address* **authentication-keyid** *keyid*

By default, no multicast server is associated with the specified key.

Controlling NTP packet sending and receiving

Specifying the source interface for NTP messages

Restrictions and guidelines

To prevent interface status changes from causing NTP communication failures, configure the device to use the IP address of an interface that is always up. For example, you can configure the device to use a loopback interface as the source IP address for the NTP messages to be sent.

When the device responds to an NTP request, the source IP address of the NTP response is always the IP address of the interface that has received the NTP request.

If you have specified the source interface for NTP messages in the **ntp-service unicast-server**/**ntp-service ipv6 unicast-server** or **ntp-service unicast-peer**/**ntp-service ipv6 unicast-peer** command, the specified interface acts as the source interface for NTP messages.

If you have configured the **ntp-service broadcast-server** or **ntp-service multicast-server**/**ntp-service ipv6 multicast-server** command in an interface view, this interface acts as the source interface for broadcast or multicast NTP messages.

Procedure

1. Enter system view.

   `system-view`

2. Specify the source interface for NTP packets.

   IPv4:

   `ntp-service source` *`interface-type interface-number`*

   IPv6:

   `ntp-service ipv6 source` *`interface-type interface-number`*

   By default, no source interface is specified for NTP messages.

Disabling an interface from receiving NTP messages

About this task

When NTP is enabled, all interfaces by default can receive NTP messages. For security purposes, you can disable some of the interfaces from receiving NTP messages.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Disable the interface from receiving NTP packets.

   IPv4:

   `undo ntp-service inbound enable`

   IPv6:

   `undo ntp-service ipv6 inbound enable`

   By default, an interface receives NTP messages.

Configuring the maximum number of dynamic associations

About this task

Perform this task to restrict the number of dynamic associations to prevent dynamic associations from occupying too many system resources.

NTP has the following types of associations:

- **Static association**—A manually created association.
- **Dynamic association**—Temporary association created by the system during NTP operation. A dynamic association is removed if no messages are exchanged within about 12 minutes.

The following describes how an association is established in different association modes:

- **Client/server mode**—After you specify an NTP server, the system creates a static association on the client. The server simply responds passively upon the receipt of a message, rather than creating an association (static or dynamic).
- **Symmetric active/passive mode**—After you specify a symmetric passive peer on a symmetric active peer, static associations are created on the symmetric active peer, and dynamic associations are created on the symmetric passive peer.
- **Broadcast or multicast mode**—Static associations are created on the server, and dynamic associations are created on the client.

Restrictions and guidelines

A single device can have a maximum of 128 concurrent associations, including static associations and dynamic associations.

Procedure

1. Enter system view.
   system-view
2. Configure the maximum number of dynamic sessions.
   ntp-service max-dynamic-sessions *number*
   By default, the maximum number of dynamic sessions is 100.

Setting a DSCP value for NTP packets

About this task

The DSCP value determines the sending precedence of an NTP packet.

Procedure

1. Enter system view.
   system-view
2. Set a DSCP value for NTP packets.
   IPv4:
   ntp-service dscp *dscp-value*
   IPv6:
   ntp-service ipv6 dscp *dscp-value*
   The default DSCP value is 48 for IPv4 packets and 56 for IPv6 packets.

Controlling output of logs and traps during time synchronization

About this task

With this feature configured, the system synchronizes the client's time to the server when the time offset exceeds 128 ms, but outputs logs and traps only when the time offset exceeds the specified thresholds, respectively.

Procedure

1. Enter system view.

   system-view

2. Set the time offset thresholds for outputting logs and traps during time synchronization.

   ntp-service time-offset-threshold { **log** *log-threshold* | **trap** *trap-threshold* } *

   By default, no time offset thresholds are set for outputting logs and traps during time synchronization.

Display and maintenance commands for NTP

Execute **display** commands in any view.

Task	Command
Display information about IPv6 NTP associations.	**display ntp-service ipv6 sessions** [**verbose**]
Display information about IPv4 NTP associations.	**display ntp-service sessions** [**verbose**]
Display information about NTP service status.	**display ntp-service status**
Display brief information about the NTP servers from the local device back to the primary NTP server.	**display ntp-service trace** [**source** *interface-type interface-number*]

Configuring SNTP

About SNTP

SNTP is a simplified, client-only version of NTP specified in RFC 4330. It uses the same packet format and packet exchange procedure as NTP, but provides faster synchronization at the price of time accuracy.

SNTP working mode

SNTP supports only the client/server mode. An SNTP-enabled device can receive time from NTP servers, but cannot provide time services to other devices.

If you specify multiple NTP servers for an SNTP client, the server with the best stratum is selected. If multiple servers are at the same stratum, the NTP server whose time packet is first received is selected.

Protocols and standards

RFC 4330, *Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI*

Restrictions and guidelines: SNTP configuration

When you configure SNTP, follow these restrictions and guidelines:

- You cannot configure both NTP and SNTP on the same device.
- You must specify a context in the **`clock protocol`** command for using NTP to obtain the time. For more information about the **`clock protocol`** command, see device management commands in *Fundamentals Command Reference*.
- You can specify only one context to use NTP for time synchronization.

SNTP tasks at a glance

To configure SNTP, perform the following tasks:

1. Enabling the SNTP service
2. Specifying an NTP server for the device
3. (Optional.) Configuring SNTP authentication

Enabling the SNTP service

Restrictions and guidelines

The NTP service and SNTP service are mutually exclusive. Before you enable SNTP, make sure NTP is disabled.

Procedure

1. Enter system view.

```
system-view
```

2. Enable the SNTP service.

   `sntp enable`

   By default, the SNTP service is disabled.

Specifying an NTP server for the device

Restrictions and guidelines

To use an NTP server as the time source, make sure its clock has been synchronized. If the stratum level of the NTP server is greater than or equal to that of the client, the client does not synchronize with the NTP server.

Procedure

1. Enter system view.

   `system-view`

2. Specify an NTP server for the device.

   IPv4:

   sntp unicast-server { *server-name* | *ip-address* } [**vpn-instance** *vpn-instance-name*] [**authentication-keyid** *keyid* | **source** *interface-type interface-number* | **version** *number*] *

   IPv6:

   sntp ipv6 unicast-server { *server-name* | *ipv6-address* } [**vpn-instance** *vpn-instance-name*] [**authentication-keyid** *keyid* | **source** *interface-type interface-number*] *

   By default, no NTP server is specified for the device.

   You can specify multiple NTP servers for the client by repeating this step.

   To perform authentication, you need to specify the **authentication-keyid** *keyid* option.

Configuring SNTP authentication

About this task

SNTP authentication ensures that an SNTP client is synchronized only to an authenticated trustworthy NTP server.

Restrictions and guidelines

Enable authentication on both the NTP server and the SNTP client.

Use the same authentication key ID and key on the NTP server and SNTP client. Specify the key as a trusted key on both the NTP server and the SNTP client. For information about configuring NTP authentication on an NTP server, see "Configuring NTP."

On the SNTP client, associate the specified key with the NTP server. Make sure the server is allowed to use the key ID for authentication on the client.

With authentication disabled, the SNTP client can synchronize with the NTP server regardless of whether the NTP server is enabled with authentication.

Procedure

1. Enter system view.

   `system-view`

2. Enable SNTP authentication.

   `sntp authentication enable`

By default, SNTP authentication is disabled.

3. Configure an SNTP authentication key.

   sntp authentication-keyid *keyid* **authentication-mode md5** { **cipher** | **simple** } *string* [**acl** *ipv4-acl-number* | **ipv6 acl** *ipv6-acl-number*] *

   By default, no SNTP authentication key exists.

4. Specify the key as a trusted key.

   sntp reliable authentication-keyid *keyid*

   By default, no trusted key is specified.

5. Associate the SNTP authentication key with an NTP server.

   IPv4:

   sntp unicast-server { *server-name* | *ip-address* } [**vpn-instance** *vpn-instance-name*] **authentication-keyid** *keyid*

   IPv6:

   sntp ipv6 unicast-server { *server-name* | *ipv6-address* } [**vpn-instance** *vpn-instance-name*] **authentication-keyid** *keyid*

   By default, no NTP server is specified.

Controlling output of logs and traps during time synchronization

About this task

With this feature configured, the system synchronizes the client's time to the server when the time offset exceeds 128 ms, but outputs logs and traps only when the time offset exceeds the specified thresholds, respectively.

Procedure

1. Enter system view.

   system-view

2. Set the time offset thresholds for outputting logs and traps during time synchronization.

   sntp time-offset-threshold { **log** *log-threshold* | **trap** *trap-threshold* } *

   By default, no time offset thresholds are set for outputting logs and traps during time synchronization.

Display and maintenance commands for SNTP

Execute **display** commands in any view.

Task	Command
Display information about all IPv6 SNTP associations.	**display sntp ipv6 sessions**
Display information about all IPv4 SNTP associations.	**display sntp sessions**

Contents

Configuring EAA

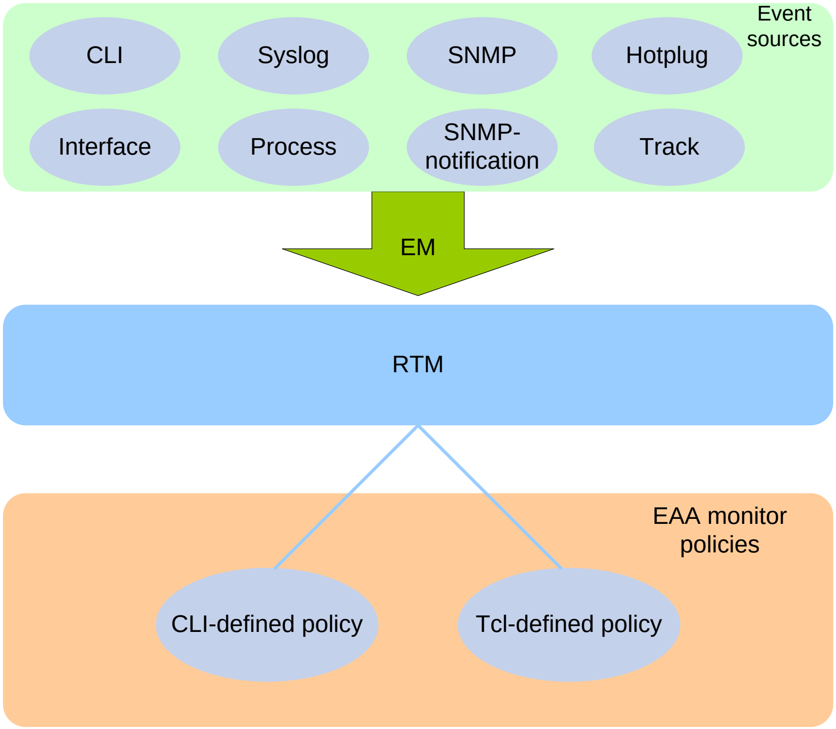
About EAA

Embedded Automation Architecture (EAA) is a monitoring framework that enables you to self-define monitored events and actions to take in response to an event. It allows you to create monitor policies by using the CLI or Tcl scripts.

EAA framework

EAA framework includes a set of event sources, a set of event monitors, a real-time event manager (RTM), and a set of user-defined monitor policies, as shown in Figure 1.

Figure 1 EAA framework



Event sources

Event sources are software or hardware modules that trigger events (see Figure 1).

For example, the CLI module triggers an event when you enter a command. The Syslog module (the information center) triggers an event when it receives a log message.

Event monitors

EAA creates one event monitor to monitor the system for the event specified in each monitor policy. An event monitor notifies the RTM to run the monitor policy when the monitored event occurs.

RTM

RTM manages the creation, state machine, and execution of monitor policies.

EAA monitor policies

A monitor policy specifies the event to monitor and actions to take when the event occurs.

You can configure EAA monitor policies by using the CLI or Tcl.

A monitor policy contains the following elements:

- One event.
- A minimum of one action.
- A minimum of one user role.
- One running time setting.

For more information about these elements, see "Elements in a monitor policy."

Elements in a monitor policy

Elements in an EAA monitor policy include event, action, user role, and runtime.

Event

Table 1 shows types of events that EAA can monitor.

Table 1 Monitored events

Event type	Description
CLI	CLI event occurs in response to monitored operations performed at the CLI. For example, a command is entered, a question mark (?) is entered, or the **Tab** key is pressed to complete a command.
Syslog	Syslog event occurs when the information center receives the monitored log within a specific period. NOTE: The log that is generated by the EAA RTM does not trigger the monitor policy to run.
Process	Process event occurs in response to a state change of the monitored process (such as an exception, shutdown, start, or restart). Both manual and automatic state changes can cause the event to occur.
Hotplug	Hot-swapping event occurs when the monitored member device joins or leaves the IRF fabric.
Interface	Each interface event is associated with two user-defined thresholds: start and restart. An interface event occurs when the monitored interface traffic statistic crosses the start threshold in the following situations: • The statistic crosses the start threshold for the first time. • The statistic crosses the start threshold each time after it crosses the restart threshold.
SNMP	Each SNMP event is associated with two user-defined thresholds: start and restart. SNMP event occurs when the monitored MIB variable's value crosses the start threshold in the following situations: • The monitored variable's value crosses the start threshold for the first time. • The monitored variable's value crosses the start threshold each time after it crosses the restart threshold.
SNMP-Notification	SNMP-Notification event occurs when the monitored MIB variable's value in an SNMP notification matches the specified condition. For example, the broadcast traffic rate on an Ethernet interface reaches or exceeds 30%.
Track	Track event occurs when the state of the track entry changes from Positive to Negative or from Negative to Positive. If you specify multiple track entries for a policy, EAA triggers the policy only when the state of all the track entries changes from Positive

Event type	Description
	(Negative) to Negative (Positive). If you set a suppress time for a policy, the timer starts when the policy is triggered. The system does not process the messages that report the track entry state change from Positive (Negative) to Negative (Positive) until the timer times out.

Action

You can create a series of order-dependent actions to take in response to the event specified in the monitor policy.

The following are available actions:

- Executing a command.
- Sending a log.
- Enabling an active/standby switchover.
- Executing a reboot without saving the running configuration.

User role

For EAA to execute an action in a monitor policy, you must assign the policy the user role that has access to the action-specific commands and resources. If EAA lacks access to an action-specific command or resource, EAA does not perform the action and all the subsequent actions.

For example, a monitor policy has four actions numbered from 1 to 4. The policy has user roles that are required for performing actions 1, 3, and 4. However, it does not have the user role required for performing action 2. When the policy is triggered, EAA executes only action 1.

For more information about user roles, see RBAC in *Fundamentals Configuration Guide*.

Runtime

The runtime limits the amount of time that the monitor policy runs its actions from the time it is triggered. This setting prevents a policy from running its actions permanently to occupy resources.

EAA environment variables

EAA environment variables decouple the configuration of action arguments from the monitor policy so you can modify a policy easily.

An EAA environment variable is defined as a <*variable_name variable_value*> pair and can be used in different policies. When you define an action, you can enter a variable name with a leading dollar sign (**$***variable_name*). EAA will replace the variable name with the variable value when it performs the action.

To change the value for an action argument, modify the value specified in the variable pair instead of editing each affected monitor policy.

EAA environment variables include system-defined variables and user-defined variables.

System-defined variables

System-defined variables are provided by default, and they cannot be created, deleted, or modified by users. System-defined variable names start with an underscore (_) sign. The variable values are set automatically depending on the event setting in the policy that uses the variables.

System-defined variables include the following types:

- **Public variable**—Available for any events.
- **Event-specific variable**—Available only for a type of event. The hotplug event-specific variable is _slot. When a member device in slot 1 joins or leaves the IRF fabric, the value of _slot is 1. When a member device in slot 2 joins or leaves the IRF fabric, the value of _slot is 2.

Table 2 shows all system-defined variables.

Table 2 System-defined EAA environment variables by event type

Event	Variable name and description
Any event	_event_id: Event ID _event_type: Event type _event_type_string: Event type description _event_time: Time when the event occurs _event_severity: Severity level of an event
CLI	_cmd: Commands that are matched
Syslog	_syslog_pattern: Log message content
Hotplug	_slot: ID of the member device that joins or leaves the IRF fabric
Interface	_ifname: Interface name
SNMP	_oid: OID of the MIB variable where an SNMP operation is performed _oid_value: Value of the MIB variable
SNMP-Notification	_oid: OID that is included in the SNMP notification.
Process	_process_name: Process name

User-defined variables

You can use user-defined variables for all types of events.

User-defined variable names can contain digits, characters, and the underscore sign (_), except that the underscore sign cannot be the leading character.

Configuring a user-defined EAA environment variable

About this task

Configure user-defined EAA environment variables so that you can use them when creating EAA monitor policies.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Configure a user-defined EAA environment variable.

   ```
   rtm environment var-name var-value
   ```

   For the system-defined variables, see Table 2.

Configuring a monitor policy

Restrictions and guidelines

Make sure the actions in different policies do not conflict. Policy execution result will be unpredictable if policies that conflict in actions are running concurrently.

You can assign the same policy name to a CLI-defined policy and a Tcl-defined policy. However, you cannot assign the same name to policies that are the same type.

A monitor policy supports only one event and runtime. If you configure multiple events for a policy, the most recent one takes effect.

A monitor policy supports a maximum of 64 valid user roles. User roles added after this limit is reached do not take effect.

Configuring a monitor policy from the CLI

Restrictions and guidelines

You can configure a series of actions to be executed in response to the event specified in a monitor policy. EAA executes the actions in ascending order of action IDs. When you add actions to a policy, you must make sure the execution order is correct. If two actions have the same ID, the most recent one takes effect.

Procedure

1. Enter system view.

   system-view

2. Create a CLI-defined policy and enter its view.

   rtm cli-policy *policy-name*

3. Configure an event for the policy.

   - Configure a CLI event.

     event cli { **async** [**skip**] | **sync** } **mode** { **execute** | **help** | **tab** } **pattern** *regular-exp*

   - Configure a hot-swapping event.

     event hotplug [**insert** | **remove**] **slot** *slot-number*

     Support for this command depends on the device model. For more information, see EAA commands in *Network Management and Monitoring Command Reference*.

   - Configure an interface event.

     event interface *interface-type interface-number* **monitor-obj** *monitor-obj* **start-op** *start-op* **start-val** *start-val* **restart-op** *restart-op* **restart-val** *restart-val* [**interval** *interval*]

   - Configure a process event.

     event process { **exception** | **restart** | **shutdown** | **start** } [**name** *process-name* [**instance** *instance-id*]] [**slot** *slot-number*]

   - Configure an SNMP event.

     event snmp oid *oid* **monitor-obj** { **get** | **next** } **start-op** *start-op* **start-val** *start-val* **restart-op** *restart-op* **restart-val** *restart-val* [**interval** *interval*]

     For the command to take effect, enable SNMP before you execute this command. The device automatically deletes this command after you disable SNMP.

   - Configure an SNMP-Notification event.

     event snmp-notification oid *oid* **oid-val** *oid-val* **op** *op* [**drop**]

     For the command to take effect, enable SNMP before you execute this command. The device automatically deletes this command after you disable SNMP.

   - Configure a Syslog event.

     event syslog priority *priority* **msg** *msg* **occurs** *times* **period** *period*

   - Configure a track event.

event track *track-list* **state { negative** | **positive** } [**suppress-time** *suppress-time*]

By default, a monitor policy does not contain an event.

If you configure multiple events for a policy, the most recent one takes effect.

4. Configure the actions to take when the event occurs.

Choose the following tasks as needed:

- Configure a CLI action.

  action *number* **cli** *command-line*
- Configure a reboot action.

  action *number* **reboot** [**slot** *slot-number*]
- Configure an active/standby switchover action.

  action *number* **switchover**
- Configure a logging action.

  action *number* **syslog priority** *priority* **facility** *local-number* **msg** *msg-body*

By default, a monitor policy does not contain any actions.

5. (Optional.) Assign a user role to the policy.

user-role *role-name*

By default, a monitor policy contains user roles that its creator had at the time of policy creation.

An EAA policy cannot have both the **security-audit** user role and any other user roles. Any previously assigned user roles are automatically removed when you assign the **security-audit** user role to the policy. The previously assigned **security-audit** user role is automatically removed when you assign any other user roles to the policy.

6. (Optional.) Configure the policy action runtime.

running-time *time*

The default policy action runtime is 20 seconds.

If you configure multiple action runtimes for a policy, the most recent one takes effect.

7. Enable the policy.

commit

By default, CLI-defined policies are not enabled.

A CLI-defined policy can take effect only after you perform this step.

Configuring a monitor policy by using Tcl

About this task

A Tcl script contains two parts: Line 1 and the other lines.

- Line 1

  Line 1 defines the event, user roles, and policy action runtime. After you create and enable a Tcl monitor policy, the device immediately parses, delivers, and executes Line 1.

  Line 1 must use the following format:

  ::platformtools::rtm::event_register *event-type arg1 arg2 arg3* … **user-role** *role-name1* | [**user-role** *role-name2* | […]] [**running-time** *running-time*]

  - The *arg1 arg2 arg3* … arguments represent event matching rules. If an argument value contains spaces, use double quotation marks ("") to enclose the value. For example, "a b c."

    - The configuration requirements for the *`event-type`*, *`user-role`*, and *`running-time`* arguments are the same as those for a CLI-defined monitor policy.
- The other lines

  From the second line, the Tcl script defines the actions to be executed when the monitor policy is triggered. You can use multiple lines to define multiple actions. The system executes these actions in sequence. The following actions are available:
    - Standard Tcl commands.
    - EAA-specific Tcl actions:
        - switchover (**`::platformtools::rtm::action switchover`**)
        - syslog (**`::platformtools::rtm::action syslog priority`** *`priority`* **`facility`** *`local-number`* **`msg`** *`msg-body`*). For more information about these arguments, see EAA commands in *Network Management and Monitoring Command Reference*.
    - Commands supported by the device.

Restrictions and guidelines

To revise the Tcl script of a policy, you must suspend all monitor policies first, and then resume the policies after you finish revising the script. The system cannot execute a Tcl-defined policy if you edit its Tcl script without first suspending these policies.

Procedure

1. Download the Tcl script file to the device by using FTP or TFTP.

   For more information about using FTP and TFTP, see *Fundamentals Configuration Guide*.
2. Create and enable a Tcl monitor policy.
   a. Enter system view.

      `system-view`

   b. Create a Tcl-defined policy and bind it to the Tcl script file.

      `rtm tcl-policy` *`policy-name tcl-filename`*

   By default, no Tcl policies exist.

   Make sure the script file is saved on all IRF member devices. This practice ensures that the policy can run correctly after a master/subordinate switchover occurs or the member device where the script file resides leaves the IRF.

Suspending monitor policies

About this task

This task suspends all CLI-defined and Tcl-defined monitor policies. If a policy is running when you perform this task, the system suspends the policy after it executes all the actions.

Restrictions and guidelines

To restore the operation of the suspended policies, execute the **`undo rtm scheduler suspend`** command.

Procedure

1. Enter system view.

   `system-view`
2. Suspend monitor policies.

   `rtm scheduler suspend`

Display and maintenance commands for EAA

Execute **display** commands except for the **display this** command in any view.

Task	Command
Display the running configuration of all CLI-defined monitor policies.	**display current-configuration**
Display user-defined EAA environment variables.	**display rtm environment** [*var-name*]
Display EAA monitor policies.	**display rtm policy** { **active** | **registered** [**verbose**] } [*policy-name*]
Display the running configuration of a CLI-defined monitor policy in CLI-defined monitor policy view.	**display this**

Contents

Monitoring and maintaining processes

About monitoring and maintaining processes

The system software of the device is a full-featured, modular, and scalable network operating system based on the Linux kernel. The system software features run the following types of independent processes:

- **User process**—Runs in user space. Most system software features run user processes. Each process runs in an independent space so the failure of a process does not affect other processes. The system automatically monitors user processes. The system supports preemptive multithreading. A process can run multiple threads to support multiple activities. Whether a process supports multithreading depends on the software implementation.
- **Kernel thread**—Runs in kernel space. A kernel thread executes kernel code. It has a higher security level than a user process. If a kernel thread fails, the system breaks down. You can monitor the running status of kernel threads.

Process monitoring and maintenance tasks at a glance

To monitor and maintain processes, perform the following tasks:

- Monitoring and maintaining user processes
  - Monitoring and maintaining processes

    The commands in this section apply to both user processes and kernel threads.
  - Monitoring and maintaining user processes

    The commands in this section apply only to user processes.
- Monitoring and maintaining kernel threads
  - Monitoring and maintaining processes

    The commands in this section apply to both user processes and kernel threads.
  - Monitoring and maintaining kernel threads

    The commands in this section apply only to kernel threads.

Monitoring and maintaining processes

About this task

The commands in this section apply to both user processes and kernel threads. You can use the commands for the following purposes:

- Display the overall memory usage.
- Display the running processes and their memory and CPU usage.
- Locate abnormal processes.

If a process consumes excessive memory or CPU resources, the system identifies the process as an abnormal process.

- If an abnormal process is a user process, troubleshoot the process as described in "Monitoring and maintaining user processes."

- If an abnormal process is a kernel thread, troubleshoot the process as described in "Monitoring and maintaining kernel threads."

Procedure

Execute the following commands in any view.

Task	Command
Display memory usage. (For more information about this command, see *Fundamentals Command Reference*.)	**display memory** [**summary**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display process state information.	**display process** [**all** | **job** *job-id* | **name** *process-name*] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display CPU usage for all processes.	**display process cpu** [**slot** *slot-number* [**cpu** *cpu-number*]]
Monitor process running state.	**monitor process** [**dumbtty**] [**iteration** *number*] [**slot** *slot-number* [**cpu** *cpu-number*]]
Monitor thread running state.	**monitor thread** [**dumbtty**] [**iteration** *number*] [**slot** *slot-number* [**cpu** *cpu-number*]]

Monitoring and maintaining user processes

About monitoring and maintaining user processes

Use this feature to monitor abnormal user processes and locate problems.

Configuring core dump

About this task

The core dump feature enables the system to generate a core dump file each time a process crashes until the maximum number of core dump files is reached. A core dump file stores information about the process. You can send the core dump files to NSFOCUS Support to troubleshoot the problems.

Restrictions and guidelines

Core dump files consume storage resources. Enable core dump only for processes that might have problems.

Procedure

Execute the following commands in user view:

1. (Optional.) Specify the directory for saving core dump files.

   exception filepath *directory*

   The directory for saving core dump files is the root directory of the default file system. For more information about the default file system, see file system management in *Fundamentals Configuration Guide*.

   This command is supported only on the default context.

2. Enable core dump for a process and specify the maximum number of core dump files, or disable core dump for a process.

process core { **maxcore** *value* | **off** } { **job** *job-id* | **name** *process-name* }

By default, a process generates a core dump file for the first exception and does not generate any core dump files for subsequent exceptions.

Display and maintenance commands for user processes

Execute **display** commands in any view and other commands in user view.

Task	Command
Display context information for process exceptions.	**display exception context** [**count** *value*] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display the core dump file directory.	**display exception filepath** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display log information for all user processes.	**display process log** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display memory usage for all user processes.	**display process memory** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display heap memory usage for a user process.	**display process memory heap job** *job-id* [**verbose**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display memory content starting from a specified memory block for a user process.	**display process memory heap job** *job-id* **address** *starting-address* **length** *memory-length* [**slot** *slot-number* [**cpu** *cpu-number*]]
Display the addresses of memory blocks with a specified size used by a user process.	**display process memory heap job** *job-id* **size** *memory-size* [**offset** *offset-size*] [**slot** *slot-number* [**cpu** *cpu-number*]]
Clear context information for process exceptions.	**reset exception context** [**slot** *slot-number* [**cpu** *cpu-number*]]

Monitoring and maintaining kernel threads

Restrictions and guidelines for monitoring and maintaining kernel threads

This command is supported only on the default context.

Configuring kernel thread deadloop detection

About this task

Kernel threads share resources. If a kernel thread monopolizes the CPU, other threads cannot run, resulting in a deadloop.

This feature enables the device to detect deadloops. If a thread occupies the CPU for a specific interval, the device determines that a deadloop has occurred, logs the event, and reboots to remove the deadloop.

Restrictions and guidelines

Change kernel thread deadloop detection settings only under the guidance of NSFOCUS Support. Inappropriate configuration can cause system breakdown. As a best practice, leave the default unchanged.

Procedure

1. Enter system view.

   system-view

2. Enable kernel thread deadloop detection.

   monitor kernel deadloop enable [**slot** *slot-number* [**cpu** *cpu-number* [**core** *core-number*&<1-64>]]]

   By default, kernel thread deadloop detection is enabled.

3. (Optional.) Set the interval for identifying a kernel thread deadloop.

   monitor kernel deadloop time *time* [**slot** *slot-number* [**cpu** *cpu-number*]]

   By default, the threshold for identifying a kernel thread deadloop is 28 seconds.

4. (Optional.) Exclude a kernel thread from kernel thread deadloop detection.

   monitor kernel deadloop exclude-thread *tid* [**slot** *slot-number* [**cpu** *cpu-number*]]

   When enabled, kernel thread deadloop detection monitors all kernel threads by default.

5. Set kernel thread deadloop protection thresholds.

   monitor kernel deadloop action threshold *threshold* [**slot** *slot-number* [**cpu** *cpu-number*]]

   By default, the kernel thread deadloop protection threshold is 1. The device takes protection actions immediately after detecting a kernel thread deadloop.

   When the number of detected kernel thread deadloops reaches the kernel thread deadloop protection threshold, the device takes protection actions to remove the deadloops.

Configuring kernel thread starvation detection

About this task

Starvation occurs when a thread is unable to access shared resources.

Kernel thread starvation detection enables the system to detect and report thread starvation. If a thread is not executed within a specific interval, the system determines that a starvation has occurred and generates a starvation message.

Thread starvation does not impact system operation. A starved thread can automatically run when certain conditions are met.

Restrictions and guidelines

Configure kernel thread starvation detection only under the guidance of NSFOCUS Support. Inappropriate configuration can cause system breakdown. As a best practice, leave the default unchanged.

Procedure

1. Enter system view.

   system-view

2. Enable kernel thread starvation detection.

   monitor kernel starvation enable [**slot** *slot-number* [**cpu** *cpu-number*]]

   By default, kernel thread starvation detection is disabled.

3. (Optional.) Set the interval for identifying a kernel thread starvation.

   monitor kernel starvation time *time* [**slot** *slot-number* [**cpu** *cpu-number*]]

   By default, the threshold for identifying a kernel thread starvation is 120 seconds.

4. (Optional.) Exclude a kernel thread from kernel thread starvation detection.

   monitor kernel starvation exclude-thread *tid* [**slot** *slot-number* [**cpu** *cpu-number*]]

   When enabled, kernel thread starvation detection monitors all kernel threads by default.

Display and maintenance commands for kernel threads

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display kernel thread deadloop detection configuration.	**display kernel deadloop configuration** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display kernel thread deadloop information.	**display kernel deadloop** *show-number* [*offset*] [**verbose**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display kernel thread exception information.	**display kernel exception** *show-number* [*offset*] [**verbose**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display kernel thread reboot information.	**display kernel reboot** *show-number* [*offset*] [**verbose**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Display kernel thread starvation detection configuration.	**display kernel starvation configuration** [**slot** *slot-number* [**cpu** *cpu-number*]]
Display kernel thread starvation information.	**display kernel starvation** *show-number* [*offset*] [**verbose**] [**slot** *slot-number* [**cpu** *cpu-number*]]
Clear kernel thread deadloop information.	**reset kernel deadloop** [**slot** *slot-number* [**cpu** *cpu-number*]]
Clear kernel thread exception information.	**reset kernel exception** [**slot** *slot-number* [**cpu** *cpu-number*]]
Clear kernel thread reboot information.	**reset kernel reboot** [**slot** *slot-number* [**cpu** *cpu-number*]]
Clear kernel thread starvation information.	**reset kernel starvation** [**slot** *slot-number* [**cpu** *cpu-number*]]

Contents

Configuring NETCONF

About NETCONF

Network Configuration Protocol (NETCONF) is an XML-based network management protocol. It provides programmable mechanisms to manage and configure network devices. Through NETCONF, you can configure device parameters, retrieve parameter values, and collect statistics. For a network that has devices from vendors, you can develop a NETCONF-based NMS system to configure and manage devices in a simple and effective way.

NETCONF structure

NETCONF has the following layers: content layer, operations layer, RPC layer, and transport protocol layer.

Table 1 NETCONF layers and XML layers

NETCONF layer	XML layer	Description
Content	Configuration data, status data, and statistics information	Contains a set of managed objects, which can be configuration data, status data, and statistics information. For information about the operable data, see the NETCONF XML API references for the device.
Operations	<get>,<get-config>, <edit-config>…	Defines a set of base operations invoked as RPC methods with XML-encoded parameters. NETCONF base operations include data retrieval operations, configuration operations, lock operations, and session operations. For the device supported operations, see "Supported NETCONF operations."
RPC	<rpc>,<rpc-reply>	Provides a simple, transport-independent framing mechanism for encoding RPCs. The <rpc> and <rpc-reply> elements are used to enclose NETCONF requests and responses (data at the operations layer and the content layer).
Transport protocol	Console/Telnet/SSH/HTTP/HTTPS/TLS	Provides reliable connection-oriented data links. The following transport layer sessions are available: • CLI sessions, including NETCONF over Telnet, NETCONF over SSH, and NETCONF over console sessions. • NETCONF over HTTP sessions and NETCONF over HTTPS sessions. • NETCONF over SOAP sessions, including NETCONF over SOAP over HTTP and NETCONF over SOAP over HTTPS sessions.

NETCONF message format

NETCONF

All NETCONF messages are XML-based and comply with RFC 4741. Any incoming NETCONF messages must pass XML Schema check before it can be processed. If a NETCONF message fails XML Schema check, the device sends an error message to the client.

For information about the NETCONF operations supported by the device and the operable data, see the NETCONF XML API reference for the device.

The following example shows a NETCONF message for getting all parameters of all interfaces on the device:

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-bulk>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces>
                 <Interface/>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get-bulk>
</rpc>
```

NETCONF over SOAP

All NETCONF over SOAP messages are XML-based and comply with RFC 4741. NETCONF messages are contained in the <Body> element of SOAP messages. NETCONF over SOAP messages also comply with the following rules:

- SOAP messages must use the SOAP Envelope namespaces.
- SOAP messages must use the SOAP Encoding namespaces.
- SOAP messages cannot contain the following information:
  - DTD reference.
  - XML processing instructions.

The following example shows a NETCONF over SOAP message for getting all parameters of all interfaces on the device:

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <auth:Authentication env:mustUnderstand="1"
xmlns:auth="http://www.nsfocus.com.cn/netconf/base:1.0">
      <auth:AuthInfo>800207F0120020C</auth:AuthInfo>
    </auth:Authentication>
  </env:Header>
  <env:Body>
    <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <get-bulk>
        <filter type="subtree">
          <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
            <Ifmgr>
              <Interfaces>
                <Interface/>
              </Interfaces>
            </Ifmgr>
          </top>
        </filter>
      </get-bulk>
    </rpc>
  </env:Body>
</env:Envelope>
```

How to use NETCONF

You can use NETCONF to manage and configure the device by using the methods in Table 2.

Table 2 NETCONF methods for configuring the device

Configuration tool	Login method	Remarks
CLI	• Console port • SSH • Telnet	To perform NETCONF operations, copy valid NETCONF messages to the CLI in XML view.
Standard Web interface for the device	• HTTP • HTTPS	The system automatically converts the configuration operations on the Web interface to NETCONF messages and sends them to the device to perform NETCONF operations.
Custom user interface	N/A	To use this method, you must enable NETCONF over SOAP. NETCONF messages will be encapsulated in SOAP for transmission.

Protocols and standards

- RFC 3339, *Date and Time on the Internet: Timestamps*
- RFC 4741, *NETCONF Configuration Protocol*
- RFC 4742, *Using the NETCONF Configuration Protocol over Secure SHell (SSH)*
- RFC 4743, *Using NETCONF over the Simple Object Access Protocol (SOAP)*
- RFC 5277, *NETCONF Event Notifications*
- RFC 5381, *Experience of Implementing NETCONF over SOAP*
- RFC 5539, *NETCONF over Transport Layer Security (TLS)*
- RFC 6241, *Network Configuration Protocol*

NETCONF tasks at a glance

To configure the device through NETCONF, perform the following tasks:

1. Configuring NETCONF to use module-specific namespaces
2. Establishing a NETCONF session
   a. (Optional.) Setting NETCONF session attributes
   b. Establishing NETCONF over SOAP sessions
   c. Establishing NETCONF over SSH sessions
   d. Establishing NETCONF over Telnet or NETCONF over console sessions
   e. Exchanging capabilities
3. (Optional.) Retrieving device configuration information
   - Retrieving device configuration and state information
   - Retrieving non-default settings
   - Retrieving NETCONF information
   - Retrieving NETCONF session information

4. (Optional.) Filtering data
   - Table-based filtering
   - Column-based filtering
5. (Optional.) Locking or unlocking the running configuration
   a. Locking the running configuration
   b. Unlocking the running configuration
6. (Optional.) Modifying the configuration
7. (Optional.) Managing configuration files
   - Saving the running configuration
   - Loading the configuration
   - Rolling back the configuration
8. (Optional.) Performing CLI operations through NETCONF
9. (Optional.) Subscribing to events
   - Subscribing to syslog events
   - Subscribing to events monitored by NETCONF
   - Subscribing to events reported by modules
10. (Optional.) Terminating NETCONF sessions
11. (Optional.) Returning to the CLI

Configuring NETCONF to use module-specific namespaces

About this task

NETCONF supports the following types of namespaces:

- **Common namespace**—The common namespace is shared by all modules. In a packet that uses the common namespace, the namespace is indicated in the <top> element, and the modules are listed under the <top> element.

  Example:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <get-bulk>
      <filter type="subtree">
         <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
            <Ifmgr>
               <Interfaces>
               </Interfaces>
            </Ifmgr>
         </top>
      </filter>
   </get-bulk>
</rpc>
```

- **Module-specific namespace**—Each module has its own namespace. A packet that uses a module-specific namespace does not have the <top> element. The namespace follows the module name.

  Example:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
   <get-bulk>
      <filter type="subtree">
            <Ifmgr xmlns="http://www.nsfocus.com.cn/netconf/data:1.0-Ifmgr">
               <Interfaces>
               </Interfaces>
            </Ifmgr>
      </filter>
   </get-bulk>
</rpc>
```

The common namespace is incompatible with module-specific namespaces. To set up a NETCONF session, the device and the client must use the same type of namespaces. By default, the common namespace is used. If the client does not support the common namespace, use this feature to configure the device to use module-specific namespaces.

Restrictions and guidelines

For this feature to take effect, you must reestablish the NETCONF session.

Procedure

1. Enter system view.
   `system-view`
2. Configure NETCONF to use module-specific namespaces.
   `netconf capability specific-namespace`
   By default, the common namespace is used.

Establishing a NETCONF session

Restrictions and guidelines for NETCONF session establishment

After a NETCONF session is established, the device automatically sends its capabilities to the client. You must send the capabilities of the client to the device before you can perform any other NETCONF operations.

Before performing a NETCONF operation, make sure no other users are configuring or managing the device. If multiple users simultaneously configure or manage the device, the result might be different from what you expect.

You can use the **`aaa session-limit`** command to set the maximum number of NETCONF sessions that the device can support. If the upper limit is reached, new NETCONF users cannot access the device. For information about this command, see AAA in *Security Configuration Guide*.

Setting NETCONF session attributes

About this task

NETCONF supports the following types of namespaces:

- **Common namespace**—The common namespace is shared by all modules. In a packet that uses the common namespace, the namespace is indicated in the <top> element, and the modules are listed under the <top> element.

  Example:

  ```
  <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  ```

```
   <get-bulk>
      <filter type="subtree">
         <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
            <Ifmgr>
               <Interfaces>
               </Interfaces>
            </Ifmgr>
         </top>
      </filter>
   </get-bulk>
</rpc>
```

- **Module-specific namespace**—Each module has its own namespace. A packet that uses a module-specific namespace does not have the <top> element. The namespace follows the module name.

  Example:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <get-bulk>
      <filter type="subtree">
            <Ifmgr xmlns="http://www.nsfocus.com.cn/netconf/data:1.0-Ifmgr">
               <Interfaces>
               </Interfaces>
            </Ifmgr>
      </filter>
   </get-bulk>
</rpc>
```

The common namespace is incompatible with module-specific namespaces. To set up a NETCONF session, the device and the client must use the same type of namespaces. By default, the common namespace is used. If the client does not support the common namespace, use this feature to configure the device to use module-specific namespaces.

Procedure

1. Enter system view.

   system-view

2. Set the NETCONF session idle timeout time.

   netconf { **agent** | **soap** } **idle-timeout** *minute*

Parameter	Description
agent	Specifies the following sessions: • NETCONF over SSH sessions. • NETCONF over Telnet sessions. • NETCONF over console sessions. By default, the idle timeout time is 0, and the sessions never time out.
soap	Specifies the following sessions: • NETCONF over SOAP over HTTP sessions. • NETCONF over SOAP over HTTPS sessions. The default setting is 10 minutes.

3. Enable NETCONF logging.

netconf log source { **all** | { **agent** | **soap** | **web** } * } { **protocol-operation** { **all** | { **action** | **config** | **get** | **set** | **session** | **syntax** | **others** } * } | **row-operation** | **verbose** }

By default, NETCONF logging is disabled.

4. Configure NETCONF to use module-specific namespaces.

   netconf capability specific-namespace

   By default, the common namespace is used.

   For the setting to take effect, you must reestablish the NETCONF session.

Establishing NETCONF over SOAP sessions

About this task

You can use a custom user interface to establish a NETCONF over SOAP session to the device and perform NETCONF operations. NETCONF over SOAP encapsulates NETCONF messages into SOAP messages and transmits the SOAP messages over HTTP or HTTPS.

Restrictions and guidelines

You can add an authentication domain to the <UserName> parameter of a SOAP request. The authentication domain takes effect only on the current request.

The mandatory authentication domain configured by using the **netconf soap domain** command takes precedence over the authentication domain specified in the <UserName> parameter of a SOAP request.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Apply an SSL server policy to the NETCONF over SOAP over HTTPS service.

   netconf soap https ssl-server-policy *policy-name*

   By default, no SSL server policy is applied to the NETCONF over SOAP over HTTPS service.

   The NETCONF over SOAP over HTTPS service will use the SSL server policy to enhance service security. For more information about SSL server policies, see SSL configuration in *Security Configuration Guide*.

   After NETCONF over SOAP over HTTPS is enabled, changes to the applied SSL server policy do not affect established NETCONF over SOAP over HTTPS sessions. The changes affect only NETCONF over SOAP over HTTPS sessions established after the changes are made.

3. Enable NETCONF over SOAP.

   netconf soap { **http** | **https** } **enable**

   By default, the NETCONF over SOAP feature is disabled.

4. (Optional.) Specify the NETCONF over SOAP over HTTP port number.

   netconf soap http port *port-number*

   By default, the NETCONF over SOAP over HTTP port number is 80.

5. (Optional.) Use an ACL to control NETCONF over SOAP access so only clients permitted by the ACL can establish NETCONF over SOAP sessions.

   netconf soap { **http** | **https** } [**ipv6**] **acl** { *acl-number* | **name** *acl-name* }

   By default, no ACL is applied to control NETCONF over SOAP access.

   The **ipv6** keyword is supported only if you have specified the **http** keyword.

6. (Optional.) Specify a mandatory authentication domain for NETCONF users.

   netconf soap domain *domain-name*

By default, no mandatory authentication domain is specified for NETCONF users. For information about authentication domains, see *Security Configuration Guide*.

7. Use the custom user interface to establish a NETCONF over SOAP session with the device.

   For information about the custom user interface, see the user guide for the interface.

Establishing NETCONF over SSH sessions

Prerequisites

Before establishing a NETCONF over SSH session, make sure the custom user interface can access the device through SSH.

Procedure

1. Enter system view.

   `system-view`

2. Enable NETCONF over SSH.

   `netconf ssh server enable`

   By default, NETCONF over SSH is disabled.

3. Specify the listening port for NETCONF over SSH packets.

   `netconf ssh server port` *`port-number`*

   By default, the listening port number is 830.

4. Use an IPv4 ACL to control NETCONF over SSH access.

   `netconf ssh acl` { *`ipv4-acl-number`* | **name** *`ipv4-acl-name`* }

   By default, no IPv4 ACL is specified to control NETCONF over SSH access.

5. Use the custom user interface to establish a NETCONF over SSH session with the device.

   For information about the custom user interface, see the user guide for the interface.

Establishing NETCONF over Telnet or NETCONF over console sessions

Restrictions and guidelines

To ensure the format correctness of a NETCONF message, do not enter the message manually. Copy and paste the message.

While the device is performing a NETCONF operation, do not perform any other operations, such as pasting a NETCONF message or pressing **Enter**.

For the device to identify NETCONF messages, you must add end mark **]]>]]>** at the end of each NETCONF message. Examples in this document do not necessarily have this end mark. Do add the end mark in actual operations.

Prerequisites

To establish a NETCONF over Telnet session or a NETCONF over console session, first log in to the device through Telnet or the console port.

Procedure

To enter XML view, execute the following command in user view:

`xml`

If the XML view prompt appears, the NETCONF over Telnet session or NETCONF over console session is established successfully.

Exchanging capabilities

About this task

After a NETCONF session is established, the device sends its capabilities to the client. You must use a hello message to send the capabilities of the client to the device before you can perform any other NETCONF operations.

Hello message from the device to the client

```
<?xml version="1.0" encoding="UTF-8"?><hello
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"><capabilities><capability>urn:ietf:pa
rams:netconf:base:1.1</capability><capability>urn:ietf:params:netconf:writable-runnin
g</capability><capability>urn:ietf:params:netconf:capability:notification:1.0</capabi
lity><capability>urn:ietf:params:netconf:capability:validate:1.1</capability><capabil
ity>urn:ietf:params:netconf:capability:interleave:1.0</capability><capability>urn:nsf
ocus:params:netconf:capability:nsfocus-netconf-ext:1.0</capability></capabilities><se
ssion-id>1</session-id></hello>]]>]]>
```

The <capabilities> element carries the capabilities supported by the device. The supported capabilities vary by device model.

The <session-id> element carries the unique ID assigned to the NETCONF session.

Hello message from the client to the device

After receiving the hello message from the device, copy the following hello message to notify the device of the capabilities supported by the client:

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
     capability-set
   </capability>
 </capabilities>
</hello>
```

Item	Description
capability-set	Specifies a set of capabilities supported by the client. Use the <capability> and </capability> tags to enclose each user-defined capability set.

Retrieving device configuration information

Restrictions and guidelines for device configuration retrieval

During a <get>, <get-bulk>, <get-config>, or <get-bulk-config> operation, NETCONF replaces unidentifiable characters in the retrieved data with question marks (?) before sending the data to the client. If the relevant module process is not started yet, the following message is sent to the client:

```
<?xml version="1.0"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data/>
</rpc-reply>
```

The <get><netconf-state/></get> operation does not support data filtering.

For more information about the NETCONF operations, see the NETCONF XML API references for the device.

Retrieving device configuration and state information

You can use the following NETCONF operations to retrieve device configuration and state information:

- **<get> operation**—Retrieves all device configuration and state information that match the specified conditions.
- **<get-bulk> operation**—Retrieves data entries starting from the data entry next to the one with the specified index. One data entry contains a device configuration entry and a state information entry. The returned output does not include the index information.

The <get> message and <get-bulk> message share the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <getoperation>
    <filter>
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
          Specify the module, submodule, table name, and column name
      </top>
    </filter>
  </getoperation>
</rpc>
```

Item	Description
getoperation	Operation name, **get** or **get-bulk**.
filter	Specifies the filtering conditions, such as the module name, submodule name, table name, and column name. • If you specify a module name, the operation retrieves the data for the specified module. If you do not specify a module name, the operation retrieves the data for all modules. • If you specify a submodule name, the operation retrieves the data for the specified submodule. If you do not specify a submodule name, the operation retrieves the data for all submodules. • If you specify a table name, the operation retrieves the data for the specified table. If you do not specify a table name, the operation retrieves the data for all tables. • If you specify only the index column, the operation retrieves the data for all columns. If you specify the index column and any other columns, the operation retrieves the data for the index column and the specified columns.

A <get-bulk> message can carry the **count** and **index** attributes.

Item	Description
index	Specifies the index. If you do not specify this item, the index value starts with 1 by default.
count	Specifies the data entry quantity. The **count** attribute complies with the following rules: • The **count** attribute can be placed in the module node and table node. In other nodes, it cannot be resolved.

Item	Description
	- When the **count** attribute is placed in the module node, a descendant node inherits this count attribute if the descendant node does not contain the count attribute. - The <get-bulk> operation retrieves all the rest data entries starting from the data entry next to the one with the specified index if either of the following conditions occurs: ○ You do not specify the **count** attribute. ○ The number of matching data entries is less than the value of the **count** attribute.

The following <get-bulk> message example specifies the **count** and **index** attributes:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xc="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get-bulk>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0"
xmlns:base="http://www.nsfocus.com.cn/netconf/base:1.0">
        <Syslog>
          <Logs xc:count="5">
            <Log>
              <Index>10</Index>
           </Log>
             </Logs>
        </Syslog>
      </top>
    </filter>
  </get-bulk>
</rpc>
```

When retrieving interface information, the device cannot identify whether an integer value for the <IfIndex> element represents an interface name or index. When retrieving VPN instance information, the device cannot identify whether an integer value for the <vrfindex> element represents a VPN name or index. To resolve the issue, you can use the **valuetype** attribute to specify the value type.

The **valuetype** attribute has the following values:

Value	Description
name	The element is carrying a name.
index	The element is carrying an index.
auto	Default value. The device uses the value of the element as a name for information matching. If no match is found, the device uses the value as an index for interface or information matching.

The following example specifies an index-type value for the <IfIndex> element:

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <getoperation>
    <filter>
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0"
xmlns:base="http://www.nsfocus.com.cn/netconf/base:1.0">
       <VLAN>
```

```
        <TrunkInterfaces>
          <Interface>
            <IfIndex base:valuetype="index">1</IfIndex>
          </Interface>
        </TrunkInterfaces>
       </VLAN>
      </top>
    </filter >
  </getoperation>
</rpc>
```

If the <get> or < get-bulk> operation succeeds, the device returns the retrieved data in the following format:

```
<?xml version="1.0"?>
 <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
     Device state and configuration data
  </data>
</rpc-reply>
```

Retrieving non-default settings

The <get-config> and <get-bulk-config> operations are used to retrieve all non-default settings. The <get-config> and <get-bulk-config> messages can contain the <filter> element for filtering data.

The <get-config> and <get-bulk-config> messages are similar. The following is a <get-config> message example:

```
<?xml version="1.0"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter>
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
          Specify the module name, submodule name, table name, and column name
      </top>
    </filter>
  </get-config>
</rpc>
```

If the <get-config> or <get-bulk-config> operation succeeds, the device returns the retrieved data in the following format:

```
<?xml version="1.0"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    Data matching the specified filter
  </data>
</rpc-reply>
```

Retrieving NETCONF information

Use the <get><netconf-state/></get> message to retrieve NETCONF information.

# Copy the following text to the client to retrieve NETCONF information:

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="m-641" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get>
      <filter type='subtree'>
        <netconf-state xmlns='urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring'>
             <getType/>
        </netconf-state>
      </filter>
    </get>
  </rpc>
```

If you do not specify a value for *getType*, the retrieval operation retrieves all NETCONF information.

The value for *getType* can be one of the following operations:

Operation	Description
capabilities	Retrieves device capabilities.
datastores	Retrieves databases from the device.
schemas	Retrieves the list of the YANG file names from the device.
sessions	Retrieves session information from the device.
statistics	Retrieves NETCONF statistics.

If the <get><netconf-state/></get> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0"?>
 <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
     Retrieved NETCONF information
  </data>
</rpc-reply>
```

Retrieving NETCONF session information

Use the <get-sessions> operation to retrieve NETCONF session information of the device.

# Copy the following message to the client to retrieve NETCONF session information from the device:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-sessions/>
</rpc>
```

If the <get-sessions> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <get-sessions>
    <Session>
      <SessionID>Configuration session ID</SessionID>
      <Line>Line information</Line>
      <UserName>Name of the user creating the session</UserName>
      <Since>Time when the session was created</Since>
      <LockHeld>Whether the session holds a lock</LockHeld>
    </Session>
  </get-sessions>
</rpc-reply>
```

Example: Retrieving a data entry for the interface table

Network configuration

Retrieve a data entry for the interface table.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
```

# Retrieve a data entry for the interface table.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:web="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-bulk>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0"
xmlns:web="http://www.nsfocus.com.cn/netconf/base:1.0">
        <Ifmgr>
          <Interfaces web:count="1">
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get-bulk>
</rpc>
```

Verifying the configuration

If the client receives the following text, the <get-bulk> operation is successful:

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:web="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
  <data>
    <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
      <Ifmgr>
        <Interfaces>
```

```
          <Interface>
            <IfIndex>3</IfIndex>
            <Name>GigabitEthernet1/0/2</Name>
            <AbbreviatedName>GE1/0/2</AbbreviatedName>
            <PortIndex>3</PortIndex>
            <ifTypeExt>22</ifTypeExt>
            <ifType>6</ifType>
            <Description>GigabitEthernet1/0/2 Interface</Description>
            <AdminStatus>2</AdminStatus>
            <OperStatus>2</OperStatus>
            <ConfigSpeed>0</ConfigSpeed>
            <ActualSpeed>100000</ActualSpeed>
            <ConfigDuplex>3</ConfigDuplex>
            <ActualDuplex>1</ActualDuplex>
          </Interface>
        </Interfaces>
      </Ifmgr>
    </top>
  </data>
</rpc-reply>
```

Example: Retrieving non-default configuration data

Network configuration

Retrieve all non-default configuration data.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Retrieve all non-default configuration data.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
  </get-config>
</rpc>
```

Verifying the configuration

If the client receives the following text, the <get-config> operation is successful:

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:web="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <data>
        <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
            <Ifmgr>
                <Interfaces>
                    <Interface>
                        <IfIndex>1307</IfIndex>
                        <Shutdown>1</Shutdown>
                    </Interface>
                    <Interface>
                        <IfIndex>1308</IfIndex>
                        <Shutdown>1</Shutdown>
                    </Interface>
                    <Interface>
                        <IfIndex>1309</IfIndex>
                        <Shutdown>1</Shutdown>
                    </Interface>
                    <Interface>
                        <IfIndex>1311</IfIndex>
                            <VlanType>2</VlanType>
                    </Interface>
                    <Interface>
                        <IfIndex>1313</IfIndex>
                            <VlanType>2</VlanType>
                    </Interface>
                </Interfaces>
            </Ifmgr>
            <Syslog>
                <LogBuffer>
                    <BufferSize>120</BufferSize>
                </LogBuffer>
            </Syslog>
            <System>
                <Device>
                    <SysName>Sysname</SysName>
                    <TimeZone>
                        <Zone>+11:44</Zone>
                        <ZoneName>beijing</ZoneName>
                    </TimeZone>
                </Device>
            </System>
        </top>
    </data>
</rpc-reply>
```

Example: Retrieving syslog configuration data

Network configuration

Retrieve configuration data for the Syslog module.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Retrieve configuration data for the Syslog module.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Syslog/>
      </top>
    </filter>
  </get-config>
</rpc>
```

Verifying the configuration

If the client receives the following text, the <get-config> operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <data>
        <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
            <Syslog>
                    <LogBuffer>
                        <BufferSize>120</BufferSize>
                    </LogBuffer>
            </Syslog>
        </top>
    </data>
</rpc-reply>
```

Example: Retrieving NETCONF session information

Network configuration

Get NETCONF session information.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Copy the following message to the client to exchange capabilities with the device:

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Copy the following message to the client to get the current NETCONF session information on the device:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-sessions/>
</rpc>
```

Verifying the configuration

If the client receives a message as follows, the operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <get-sessions>
        <Session>
            <SessionID>1</SessionID>
            <Line>vty0</Line>
            <UserName></UserName>
            <Since>2017-01-07T00:24:57</Since>
            <LockHeld>false</LockHeld>
        </Session>
    </get-sessions>
</rpc-reply>
```

The output shows the following information:

- The session ID of an existing NETCONF session is 1.
- The login user type is vty0.
- The login time is 2017-01-07T00:24:57.
- The user does not hold the lock of the configuration.

Filtering data

About data filtering

You can define a filter to filter information when you perform a <get>, <get-bulk>, <get-config>, or <get-bulk-config> operation. Data filtering includes the following types:

- **Table-based filtering**—Filters table information.
- **Column-based filtering**—Filters information for a single column.

Restrictions and guidelines for data filtering

For table-based filtering to take effect, you must configure table-based filtering before column-based filtering.

Table-based filtering

About this task

The namespace is **http://www.nsfocus.com.cn/netconf/base:1.0**. The attribute name is **filter**. For information about the support for table-based match, see the NETCONF XML API references.

# Copy the following text to the client to retrieve the longest data with IP address **1.1.1.0** and mask length **24** from the IPv4 routing table:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Route>
         <Ipv4Routes>
           <RouteEntry nsfocus:filter="IP 1.1.1.0 MaskLen 24 longer"/>
         </Ipv4Routes>
        </Route>
      </top>
    </filter>
  </get>
</rpc>
```

Restrictions and guidelines

To use table-based filtering, specify a match criterion for the **filter** row attribute.

Column-based filtering

About this task

Column-based filtering includes full match filtering, regular expression match filtering, and conditional match filtering. Full match filtering has the highest priority and conditional match filtering has the lowest priority. When more than one filtering criterion is specified, the one with the highest priority takes effect.

Full match filtering

You can specify an element value in an XML message to implement full match filtering. If multiple element values are provided, the system returns the data that matches all the specified values.

# Copy the following text to the client to retrieve configuration data of all interfaces in UP state:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface>
              <AdminStatus>1</AdminStatus>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get>
</rpc>
```

You can also specify an attribute name that is the same as a column name of the current table at the row to implement full match filtering. The system returns only configuration data that matches this attribute name. The XML message equivalent to the above element-value-based full match filtering is as follows:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <top
xmlns="http://www.nsfocus.com.cn/netconf/data:1.0"xmlns:data="http://www.nsfocus.com.
cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface data:AdminStatus="1"/>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get>
</rpc>
```

The above examples show that both element-value-based full match filtering and attribute-name-based full match filtering can retrieve the same index and column information for all interfaces in up state.

Regular expression match filtering

To implement a complex data filtering with characters, you can add a **regExp** attribute for a specific element.

The supported data types include integer, date and time, character string, IPv4 address, IPv4 mask, IPv6 address, MAC address, OID, and time zone.

# Copy the following text to the client to retrieve the descriptions of interfaces, of which all the characters must be upper-case letters from A to Z:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface>
              <Description nsfocus:regExp="^[A-Z]*$"/>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get-config>
</rpc>
```

Conditional match filtering

To implement a complex data filtering with digits and character strings, you can add a **match** attribute for a specific element. Table 3 lists the conditional match operators.

Table 3 Conditional match operators

Operation	**Operator**	**Remarks**
More than	match="more:*value*"	More than the specified value. The supported data types include date, digit, and character string.
Less than	match="less:*value*"	Less than the specified value. The supported data types include date, digit, and character string.
Not less than	match="notLess:*value*"	Not less than the specified value. The supported data types include date, digit, and character string.
Not more than	match="notMore:*value*"	Not more than the specified value. The supported data types include date, digit, and character string.
Equal	match="equal:*value*"	Equal to the specified value. The supported data types include date, digit, character string, OID, and BOOL.
Not equal	match="notEqual:*value*"	Not equal to the specified value. The supported data types include date, digit, character string, OID, and BOOL.
Include	match="include:*string*"	Includes the specified string. The supported data types include only character string.
Not include	match="exclude:*string*"	Excludes the specified string. The supported data types include only character string.
Start with	match="startWith:*string*"	Starts with the specified string. The supported data types include character string and OID.
End with	match="endWith:*string*"	Ends with the specified string. The supported data types include only character string.

# Copy the following text to the client to retrieve extension information about the entity whose CPU usage is more than 50%:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Device>
          <ExtPhysicalEntities>
            <Entity>
              <CpuUsage nsfocus:match="more:50"></CpuUsage>
            </Entity>
          </ExtPhysicalEntities>
        </Device>
      </top>
    </filter>
  </get>
</rpc>
```

Example: Filtering data with regular expression match

Network configuration

Retrieve all data including **Gigabit** in the **Description** column of the Interfaces table under the Ifmgr module.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Retrieve all data including **Gigabit** in the **Description** column of the Interfaces table under the Ifmgr module.

```
<?xml version="1.0"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface>
              <Description nsfocus:regExp="(Gigabit)+"/>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get>
</rpc>
```

Verifying the configuration

If the client receives the following text, the operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0" message-id="100">
    <data>
        <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
            <Ifmgr>
                <Interfaces>
                    <Interface>
                        <IfIndex>2681</IfIndex>
                        <Description>GigabitEthernet1/0/1 Interface</Description>
                    </Interface>
                    <Interface>
                        <IfIndex>2685</IfIndex>
                        <Description>GigabitEthernet1/0/2 Interface</Description>
                    </Interface>
                    <Interface>
                        <IfIndex>2689</IfIndex>
                        <Description>GigabitEthernet1/0/3 Interface</Description>
                    </Interface>
                 <Interface>
            </Ifmgr>
        </top>
    </data>
</rpc-reply>
```

Example: Filtering data by conditional match

Network configuration

Retrieve data in the **Name** column with the ifindex value not less than 5000 in the Interfaces table under the Ifmgr module.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Retrieve data in the **Name** column with the ifindex value not less than 5000 in the Interfaces table under the Ifmgr module.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface>
              <IfIndex nsfocus:match="notLess:5000"/>
              <Name/>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get>
</rpc>
```

Verifying the configuration

If the client receives the following text, the operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0" message-id="100">
    <data>
        <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
            <Ifmgr>
                <Interfaces>
                    <Interface>
                        <IfIndex>7241</IfIndex>
                        <Name>NULL0</Name>
                    </Interface>
                </Interfaces>
            </Ifmgr>
        </top>
    </data>
</rpc-reply>
```

Locking or unlocking the running configuration

About configuration locking and unlocking

Multiple methods are available for configuring the device, such as CLI, NETCONF, and SNMP. Before configuring, managing, or troubleshooting the device, you can lock the configuration to prevent other users from changing the device configuration. After you lock the configuration, only you can perform <edit-config> operations to change the configuration or unlock the configuration. Other users can only read the configuration.

If you close your NETCONF session, the system unlocks the configuration. You can also manually unlock the configuration.

Restrictions and guidelines for configuration locking and unlocking

The <lock> operation locks the running configuration of the device. You cannot use it to lock the configuration for a specific module.

Locking the running configuration

# Copy the following text to the client to lock the running configuration:

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <lock>
      <target>
        <running/>
      </target>
    </lock>
  </rpc>
```

If the <lock> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
     <ok/>
</rpc-reply>
```

Unlocking the running configuration

# Copy the following text to the client to unlock the running configuration:

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <unlock>
      <target>
        <running/>
      </target>
    </unlock>
  </rpc>
```

If the <unlock> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Example: Locking the running configuration

Network configuration

Lock the device configuration so other users cannot change the device configuration.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <capabilities>
        <capability>
            urn:ietf:params:netconf:base:1.0
        </capability>
    </capabilities>
</hello>
```

# Lock the configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <lock>
      <target>
        <running/>
      </target>
    </lock>
  </rpc>
```

Verifying the configuration

If the client receives the following response, the <lock> operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

If another client sends a lock request, the device returns the following response:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<rpc-error>
  <error-type>protocol</error-type>
  <error-tag>lock-denied</error-tag>
  <error-severity>error</error-severity>
  <error-message xml:lang="en"> Lock failed because the NETCONF lock is held by another
session.</error-message>
  <error-info>
    <session-id>1</session-id>
  </error-info>
  </rpc-error>
</rpc-reply>
```

The output shows that the <lock> operation failed. The client with session ID 1 is holding the lock,

Modifying the configuration

About the <edit-config> operation

The <edit-config> operation includes the following operations: merge, create, replace, remove, delete, default-operation, error-option, test-option, and incremental. For more information about the operations, see "Supported NETCONF operations."

Procedure

# Copy the following text to perform the <edit-config> operation:

```
<?xml version="1.0"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target><running></running></target>
    <error-option>
       error-option
    </error-option>
    <config>
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        Specify the module name, submodule name, table name, and column name
      </top>
    </config>
  </edit-config>
</rpc>
```

The <error-option> element indicates the action to be taken in response to an error that occurs during the operation. It has the following values:

Value	Description
stop-on-error	Stops the <edit-config> operation.
continue-on-error	Continues the <edit-config> operation.
rollback-on-error	Rolls back the configuration to the configuration before the <edit-config> operation was performed. By default, an <edit-config> operation cannot be performed while the device is rolling back the configuration. If the rollback time exceeds the maximum time that the client can wait, the client determines that the <edit-config> operation has failed and performs the operation again. Because the previous rollback is not completed, the operation triggers another rollback. If this process repeats itself, CPU and memory resources will be exhausted and the device will reboot. To allow an <edit-config> operation to be performed during a configuration rollback, perform an <action> operation to change the value of the **DisableEditConfigWhenRollback** attribute to **false**.

If the <edit-config> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0">
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
    </rpc-reply>
```

You can also perform the <get> operation to verify that the current element value is the same as the value specified through the <edit-config> operation.

Example: Modifying the configuration

Network configuration

Change the log buffer size for the Syslog module to 512.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
```

# Change the log buffer size for the Syslog module to 512.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:web="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0" web:operation="merge">
        <Syslog>
          <LogBuffer>
            <BufferSize>512</BufferSize>
          </LogBuffer>
        </Syslog>
      </top>
    </config>
  </edit-config>
</rpc>
```

Verifying the configuration

If the client receives the following text, the <edit-config> operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Saving the running configuration

About the <save> operation

A <save> operation saves the running configuration to a configuration file and specifies the file as the main next-startup configuration file.

Restrictions and guidelines

The <save> operation is resource intensive. Do not perform this operation when system resources are heavily occupied.

Procedure

# Copy the following text to the client:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save OverWrite="false" Binary-only="false">
    <file>Configuration file name</file>
  </save>
</rpc>
```

Item	Description
OverWrite	Determines whether to overwrite the specified file if the file already exists. Available values: • **true**—Overwrites the file. • **false**—Does not overwrite the file. If a file with the same name already exists, the system returns an error message without saving the running configuration. The default value is **true**.
Binary-only	Determines whether to save the running configuration only to the binary configuration file to accelerate the saving process. Available values: • **true**—Saves the running configuration only to the binary configuration file. ○ If you specify a nonexistent configuration file for the **file** attribute, the <save> operation fails. ○ If you do not specify the **file** attribute, the device searches for the .cfg next-startup configuration file. If the file exists, the device saves the running configuration to its corresponding binary file. If the file does not exist, the <save> operation fails. • **false**—Save the running configuration to both the .cfg text configuration file (.cfg file) and .mdb binary configuration file. Saving the running configuration only to the binary configuration file requires less time than saving the running configuration to both the text and binary configuration files. The default value is **false**.
file	Specifies a .cfg configuration file by its name or by its path. If you specify a file by its path, make sure you specify the absolute path of the file. If you include the **file** attribute in the RPC message, you must specify a configuration file. If the specified configuration file does not exist, the device creates binary and text configuration files to save the running configuration.

Item	Description
	If you do not include the **file** attribute in the RPC message, the device saves the running configuration to the text and binary next-startup configuration files.

If the <save> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Example: Saving the running configuration

Network configuration

Save the running configuration to the **config.cfg** file.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Save the running configuration of the device to the **config.cfg** file.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save>
    <file>config.cfg</file>
  </save>
</rpc>
```

Verifying the configuration

If the client receives the following response, the <save> operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Loading the configuration

About the <load> operation

The <load> operation merges the configuration from a configuration file into the running configuration as follows:

- Loads settings that do not exist in the running configuration.
- Overwrites settings that already exist in the running configuration.

Restrictions and guidelines

When you perform a <load> operation, follow these restrictions and guidelines:

- The <load> operation is resource intensive. Do not perform this operation when the system resources are heavily occupied.
- Some settings in a configuration file might conflict with the existing settings. For the settings in the file to take effect, delete the existing conflicting settings, and then load the configuration file.

Procedure

# Copy the following text to the client to load a configuration file for the device:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <load>
   <file>Configuration file name</file>
  </load>
</rpc>
```

The configuration file name must start with the storage media name and end with the **.cfg** extension.

If the <load> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Rolling back the configuration

Restrictions and guidelines

The <rollback> operation is resource intensive. Do not perform this operation when the system is busy.

By default, an <edit-config> operation cannot be performed while the device is rolling back the configuration. To allow an <edit-config> operation to be performed during a configuration rollback, perform an <action> operation to change the value of the **DisableEditConfigWhenRollback** attribute to **false**.

Rolling back the configuration based on a configuration file

# Copy the following text to the client to roll back the running configuration to the configuration in a configuration file:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rollback>
    <file>Specify the configuration file name</file>
  </rollback>
</rpc>
```

If the <rollback> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Rolling back the configuration based on a rollback point

About this task

You can roll back the running configuration based on a rollback point when one of the following situations occurs:

- A NETCONF client sends a rollback request.
- The NETCONF session idle time is longer than the rollback idle timeout time.
- A NETCONF client is unexpectedly disconnected from the device.

Restrictions and guidelines

Multiple users might simultaneously configure the device. As a best practice, lock the system before rolling back the configuration to prevent other users from modifying the running configuration.

Procedure

1. Lock the running configuration. For more information, see "Locking or unlocking the running configuration."
2. Enable configuration rollback based on a rollback point.

   # Copy the following text to the client to perform a <save-point>/<begin> operation:

   ```
   <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
         <save-point>
           <begin>
             <confirm-timeout>100</confirm-timeout>
          </begin>
         </save-point>
   </rpc>
   ```

Item	Description
confirm-timeout	Specifies the rollback idle timeout time in the range of 1 to 65535 seconds. The default is 600 seconds. This item is optional.

   If the <save-point/begin> operation succeeds, the device returns a response in the following format:

   ```
   <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
     <data>
       <save-point>
          <commit>
             <commit-id>1</commit-id>
          </commit>
       </save-point>
     </data>
   </rpc-reply>
   ```
3. Modify the running configuration. For more information, see "Modifying the configuration."
4. Mark the rollback point.

The system supports a maximum of 50 rollback points. If the limit is reached, specify the **force** attribute for the <save-point>/<commit> operation to overwrite the earliest rollback point.

# Copy the following text to the client to perform a <save-point>/<commit> operation:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <commit>
      <label>SUPPORT VLAN<label>
      <comment>vlan 1 to 100 and interfaces.</comment>
     </commit>
  </save-point>
</rpc>
```

The <label> and <comment> elements are optional.

If the <save-point>/<commit> operation succeeds, the device returns a response in the following format:

```
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <save-point>
       <commit>
          <commit-id>2</commit-id>
       </commit>
    </save-point>
  </data>
</rpc-reply>
```

5. Retrieve the rollback point configuration records.

The following text shows the message format for a <save-point/get-commits> request:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <get-commits>
      <commit-id/>
      <commit-index/>
      <commit-label/>
    </get-commits>
  </save-point>
</rpc>
```

Specify the <commit-id/>, <commit-index/>, or <commit-label/> element to retrieve the specified rollback point configuration records. If no element is specified, the operation retrieves records for all rollback point settings.

# Copy the following text to the client to perform a <save-point>/<get-commits> operation:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <get-commits>
      <commit-label>SUPPORT VLAN</commit-label>
    </get-commits>
  </save-point>
</rpc>
```

If the <save-point/get-commits> operation succeeds, the device returns a response in the following format:

```
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <data>
      <save-point>
         <commit-information>
```

```
           <CommitID>2</CommitID>
           <TimeStamp>Sun Jan 1 11:30:28 2017</TimeStamp>
           <UserName>test</UserName>
           <Label>SUPPORT VLAN</Label>
         </commit-information>
    </save-point>
  </data>
</rpc-reply>
```

6. Retrieve the configuration data corresponding to a rollback point.

   The following text shows the message format for a <save-point>/<get-commit-information> request:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <get-commit-information>
       <commit-information>
         <commit-id/>
         <commit-index/>
         <commit-label/>
      </commit-information>
      <compare-information>
         <commit-id/>
         <commit-index/>
         <commit-label/>
      </compare-information>
    </get-commit-information>
  </save-point>
</rpc>
```

   Specify one of the following elements: <commit-id/>, <commit-index/>, and <commit-label/>. The <compare-information> element is optional.

Item	Description
commit-id	Uniquely identifies a rollback point.
commit-index	Specifies 50 most recently configured rollback points. The value of 0 indicates the most recently configured one and 49 indicates the earliest configured one.
commit-label	Specifies a unique label for a rollback point.
get-commit-information	Retrieves the configuration data corresponding to the most recently configured rollback point.

   # Copy the following text to the client to perform a <save-point>/<get-commit-information> operation:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <get-commit-information>
               <commit-information>
                  <commit-label>SUPPORT VLAN</commit-label>
               </commit-information>
    </get-commit-information>
  </save-point>
</rpc>
```

If the <save-point/get-commit-information> operation succeeds, the device returns a response in the following format:

```
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <data>
     <save-point>
        <commit-information>
           <content>
              …
              interface vlan 1
              …
           </content>
        </commit-information>
     </save-point>
   </data>
</rpc-reply>
```

7. Roll back the configuration based on a rollback point.

   The configuration can also be automatically rolled back based on the most recently configured rollback point when the NETCONF session idle timer expires.

   # Copy the following text to the client to perform a <save-point>/<rollback> operation:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <rollback>
      <commit-id/>
      <commit-index/>
      <commit-label/>
    </rollback>
  </save-point>
</rpc>
```

   Specify one of the following elements: <commit-id/>, <commit-index/>, and <commit-label/>. If no element is specified, the operation rolls back configuration based on the most recently configured rollback point.

Item	Description
commit-id	Uniquely identifies a rollback point.
commit-index	Specifies 50 most recently configured rollback points. The value of 0 indicates the most recently configured one and 49 indicates the earliest configured one.
commit-label	Specifies the unique label of a rollback point.

   If the <save-point/rollback> operation succeeds, the device returns a response in the following format:

```
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok></ok>
</rpc-reply>
```

8. End the rollback configuration.

   # Copy the following text to the client to perform a <save-point>/<end> operation:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save-point>
    <end/>
  </save-point>
</rpc>
```

If the <save-point/end> operation succeeds, the device returns a response in the following format:

```
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

9. Unlock the configuration. For more information, see "Locking or unlocking the running configuration."

Performing CLI operations through NETCONF

About CLI operations through NETCONF

You can enclose command lines in XML messages to configure the device.

Restrictions and guidelines

Performing CLI operations through NETCONF is resource intensive. As a best practice, do not perform the following tasks:

- Enclose multiple command lines in one XML message.
- Use NETCONF to perform a CLI operation when other users are performing NETCONF CLI operations.

Procedure

# Copy the following text to the client to execute the commands:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <CLI>
    <Execution>
      Commands
    </Execution>
  </CLI>
</rpc>
```

The <Execution> element can contain multiple commands, with one command on one line.

If the CLI operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <CLI>
    <Execution>
      <![CDATA[Responses to the commands]]>
    </Execution>
  </CLI>
</rpc-reply>
```

Example: Performing CLI operations

Network configuration

Send the **`display vlan`** command to the device.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Copy the following text to the client to execute the **`display vlan`** command:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <CLI>
    <Execution>
          display vlan
    </Execution>
  </CLI>
</rpc>
```

Verifying the configuration

If the client receives the following text, the operation is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <CLI>
    <Execution><![CDATA[
<Sysname>display vlan
 Total VLANs: 1
 The VLANs include:
 1(default)
   ]]>
   </Execution>
  </CLI>
</rpc-reply>
```

Subscribing to events

About event subscription

When an event takes place on the device, the device sends information about the event to NETCONF clients that have subscribed to the event.

Restrictions and guidelines

The device does not support subscribing to events for NETCONF over SOAP sessions.

A subscription takes effect only on the current session. It is canceled when the session is terminated.

If you do not specify the event stream to be subscribed, the device sends syslog event notifications to the NETCONF client.

Subscribing to syslog events

# Copy the following message to the client to complete the subscription:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <create-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>NETCONF</stream>
      <filter>
        <event xmlns="http://www.nsfocus.com.cn/netconf/event:1.0">
          <Code>code</Code>
            <Group>group</Group>
              <Severity>severity</Severity>
        </event>
      </filter>
      <startTime>start-time</startTime>
      <stopTime>stop-time</stopTime>
   </create-subscription>
</rpc>
```

Item	Description
stream	Specifies the event stream. The name for the syslog event stream is **NETCONF**.
event	Specifies the event. For information about the events to which you can subscribe, see the system log message references for the device.
code	Specifies the mnemonic symbol of the log message.
group	Specifies the module name of the log message.
severity	Specifies the severity level of the log message.
start-time	Specifies the start time of the subscription.
stop-time	Specifies the end time of the subscription.

If the subscription succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

If the subscription fails, the device returns an error message in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
<rpc-error>
   <error-type>error-type</error-type>
   <error-tag>error-tag</error-tag>
   <error-severity>error-severity</error-severity>
   <error-message xml:lang="en">error-message</error-message>
</rpc-error>
</rpc-reply>
```

For more information about error messages, see RFC 4741.

Subscribing to events monitored by NETCONF

After you subscribe to events as described in this section, NETCONF regularly polls the subscribed events and sends the events that match the subscription condition to the NETCONF client.

# Copy the following message to the client to complete the subscription:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<create-subscription xmlns='urn:ietf:params:xml:ns:netconf:notification:1.0'>
  <stream>NETCONF_MONITOR_EXTENSION</stream>
  <filter>
    <NetconfMonitor xmlns='http://www.nsfocus.com.cn/netconf/monitor:1.0'>
      <XPath>XPath</XPath>
      <Interval>interval</Interval>
      <ColumnConditions>
        <ColumnCondition>
          <ColumnName>ColumnName</ColumnName>
          <ColumnValue>ColumnValue</ColumnValue>
          <ColumnCondition>ColumnCondition</ColumnCondition>
        </ColumnCondition>
      </ColumnConditions>
      <MustIncludeResultColumns>
        <ColumnName>columnName</ColumnName>
      </MustIncludeResultColumns>
    </NetconfMonitor>
  </filter>
<startTime>start-time</startTime>
<stopTime>stop-time</stopTime>
</create-subscription>
</rpc>
```

Item	Description
stream	Specifies the event stream. The name for the event stream is **NETCONF_MONITOR_EXTENSION**.
NetconfMonitor	Specifies the filtering information for the event.
XPath	Specifies the path of the event in the format of *ModuleName*[/*SubmoduleName*]/*TableName*.
interval	Specifies the interval for NETCONF to obtain events that matches the subscription condition. The value range is 1 to 4294967 seconds. The default

Item	Description
	value is 300 seconds.
ColumnName	Specifies the name of a column in the format of [*GroupName*.]*ColumnName*.
ColumnValue	Specifies the baseline value.
ColumnCondition	Specifies the operator: • **more**. • **less**. • **notLess**. • **notMore**. • **equal**. • **notEqual**. • **include**. • **exclude**. • **startWith**. • **endWith**. Choose an operator according to the type of the baseline value.
start-time	Specifies the start time of the subscription.
stop-time	Specifies the end time of the subscription.

If the subscription succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Subscribing to events reported by modules

After you subscribe to events as described in this section, the specified modules report subscribed events to NETCONF. NETCONF sends the events to the NETCONF client.

# Copy the following message to the client to complete the subscription:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xs="http://www.nsfocus.com.cn/netconf/base:1.0">
<create-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<stream>XXX_STREAM</stream>
   <filter type="subtree">
<event xmlns="http://www.nsfocus.com.cn/netconf/event:1.0/xxx-features-list-name:1.0">
       <ColumnName xs:condition="Condition">value</ColumnName>
</event>
</filter>
<startTime>start-time</startTime>
<stopTime>stop-time</stopTime>
</create-subscription>
</rpc>
```

Item	Description
`stream`	Specifies the event stream. Supported event streams vary by device model.
`event`	Specifies the event name. An event stream includes multiple events. The events use the same namespaces as the event stream.
`ColumnName`	Specifies the name of a column.
`Condition`	Specifies the operator: • **more**. • **less**. • **notLess**. • **notMore**. • **equal**. • **notEqual**. • **include**. • **exclude**. • **startWith**. • **endWith**. Choose an operator according to the type of the baseline value.
`value`	Specifies the baseline value for the column.
`start-time`	Specifies the start time of the subscription.
`stop-time`	Specifies the end time of the subscription.

If the subscription succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Canceling an event subscription

# Copy the following message to the client to cancel a subscription:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <cancel-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>XXX_STREAM</stream>
  </cancel-subscription>
</rpc>
```

Item	Description
`stream`	Specifies the event stream.

If the cancelation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <ok/>
</rpc-reply>
```

If the subscription to be canceled does not exist, the device returns an error message in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<rpc-error>
   <error-type>error-type</error-type>
   <error-tag>error-tag</error-tag>
   <error-severity>error-severity</error-severity>
<error-message xml:lang="en">The subscription stream to be canceled doesn't exist: Stream
name=XXX_STREAM.</error-message>
</rpc-error>
</rpc-reply>
```

Example: Subscribing to syslog events

Network configuration

Configure a client to subscribe to syslog events with no time limitation. After the subscription is successful, all events on the device are sent to the client before the session between the device and client is terminated.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Subscribe to syslog events with no time limitation.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <create-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>NETCONF</stream>
  </create-subscription>
</rpc>
```

Verifying the configuration

# If the client receives the following response, the subscription is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <ok/>
</rpc-reply>
```

# When another client (192.168.100.130) logs in to the device, the device sends a notification to the client that has subscribed to all events:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2011-01-04T12:30:52</eventTime>
    <event xmlns="http://www.nsfocus.com.cn/netconf/event:1.0">
        <Group>SHELL</Group>
        <Code>SHELL_LOGIN</Code>
        <Slot>1</Slot>
        <Severity>Notification</Severity>
        <context>VTY logged in from 192.168.100.130.</context>
    </event>
</notification>
```

Example: Subscribing to events reported by modules

Network configuration

Configure a client to subscribe to InterfaceEvent events reported by the interface management module. After the subscription is successful, all relevant InterfaceEvent events are sent to the client before the session between the device and client is terminated.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
      urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Subscribe to InterfaceEvent events with no time limitation.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xs="http://www.nsfocus.com.cn/netconf/base:1.0">
  <create-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>Ifmgr</stream>
    <filter type="subtree">
      <InterfaceEvent xmlns="http://www.nsfocus.com.cn/netconf/event:1.0/Ifmgr:1.0">
      </InterfaceEvent>
    </filter>
  </create-subscription>
</rpc>
```

Verifying the configuration

# If the client does not have the privilege for the operation, the following response is displayed on the client:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xs="http://www.nsfocus.com.cn/netconf/base:1.0" message-id="100">
    <rpc-error xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <error-type>application</error-type>
        <error-tag>access-denied</error-tag>
        <error-severity>error</error-severity>
        <error-message xml:lang="en">You don't have the privilege to subscribe the InterfaceEvent event.</error-message>
    </rpc-error>
</rpc-reply>
```

# If the client receives the following response, the subscription is successful:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="100">
    <ok/>
</rpc-reply>
```

# When the status of an Ethernet interface changes, the client receives the following response:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2019-04-28T17:31:03</eventTime>
    <InterfaceEvent xmlns="Ifmgr:1.0">
    <Interface>
        <Name>GigabitEthernet1/0/1</Name>
        <Status>IF_ACTIVE</Status>
        <IfIndex>261</IfIndex>
        <AdminStatus>ADMIN_UP</AdminStatus>
        <OperStatus>OPER_DOWN</OperStatus>
        <Description>The Interface GigabitEthernet1/0/1 occurred IF_ACTIVE event,the administration status is ADMIN_UP,operation status is OPER_DOWN.</Description>
    </Interface>
    </InterfaceEvent>
</notification>
```

Terminating NETCONF sessions

About NETCONF session termination

NETCONF allows one client to terminate the NETCONF sessions of other clients. A client whose session is terminated returns to user view.

Procedure

# Copy the following message to the client to terminate a NETCONF session:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <kill-session>
    <session-id>
      Specified session-ID
    </session-id>
  </kill-session>
</rpc>
```

If the <kill-session> operation succeeds, the device returns a response in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
```

```
</rpc-reply>
```

Example: Terminating another NETCONF session

Network configuration

The user whose session's ID is 1 terminates the session with session ID 2.

Procedure

# Enter XML view.

```
<Sysname> xml
```

# Notify the device of the NETCONF capabilities supported on the client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
            urn:ietf:params:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

# Terminate the session with session ID 2.

```
<rpc message-id="100"xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <kill-session>
    <session-id>2</session-id>
  </kill-session>
</rpc>
```

Verifying the configuration

If the client receives the following text, the NETCONF session with session ID 2 has been terminated, and the client with session ID 2 has returned from XML view to user view:

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Returning to the CLI

Restrictions and guidelines

Once you enter XML view, you must complete capability exchange between the device and the client before you can return to the CLI.

Procedure

# Copy the following text to the client to return from XML view to the CLI:

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <close-session/>
</rpc>
```

When the device receives the close-session request, it sends the following response and returns to CLI's user view:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Example: Returning to the CLI

Network configuration

Return to the CLI after entering XML view.

Procedure

1. Perform capability exchange between the device and the client.

   Copy the following text to the client:

   ```
   <hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
     <capabilities>
       <capability>urn:ietf:params:netconf:base:1.0</capability>
     </capabilities>
   </hello>]]>]]>
   ```

2. Return to the CLI.

   Copy the following text to the client:

   ```
   <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
     <close-session/>
   </rpc>]]>]]>
   ```

Verifying the configuration

# Verify that the <Sysname> prompt appears.

```
<Sysname>
```

Display and maintenance commands for NETCONF

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display current NETCONF service status and global NETCONF service statistics.	**display netconf service**
Display NETCONF session status and statistics.	**display netconf session**
Clear NETCONF service statistics.	**reset netconf service statistics**
Clear NETCONF session statistics.	**reset netconf session statistics**

Supported NETCONF operations

This chapter describes NETCONF operations available with NF

action

Usage guidelines

This operation issues actions for non-default settings, for example, reset action.

XML example

# Clear statistics information for all interfaces.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <action>
    <top xmlns="http://www.nsfocus.com.cn/netconf/action:1.0">
      <Ifmgr>
            <ClearAllIfStatistics>
                <Clear>
                </Clear>
        </ClearAllIfStatistics>
      </Ifmgr>
    </top>
  </action>
</rpc>
```

CLI

Usage guidelines

This operation executes CLI commands.

A request message encloses commands in the <CLI> element. A response message encloses the command output in the <CLI> element.

You can use the following elements to execute commands:

- **Execution**—Executes commands in user view.
- **Configuration**—Executes commands in system view.

  To use this element, include the **exec-use-channel** attribute and specify a value for the attribute:

  - **false**—Executes commands without using a channel.
  - **true**—Executes commands by using a temporary channel. The channel is automatically closed after the execution.
  - **persist**—Executes commands by using the persistent channel for the session.

    To use the persistent channel, first perform an <Open-channel> operation to open the persistent channel. If you do not do so, the system will automatically open the persistent channel.

    After using the persistent channel, perform a <Close-channel> operation to close the channel and return to system view. If you do not perform an <Open-channel> operation, the system stays in the view and will execute subsequent commands in the view.

NOTE:

To execute commands in a lower-level view of the system view, use the <Configuration> element to enter the view first.

A NETCONF session can have multiple temporary channels, but it can have only one persistent channel.

NETCONF does not support executing interactive commands.

You cannot execute the **quit** command by using a channel to exit user view.

XML example

# Execute the **display this** command in system view without using a channel.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <CLI>
     <Configuration exec-use-channel="false">display this</Configuration>
  </CLI>
</rpc>
```

close-session

Usage guidelines

This operation terminates the current NETCONF session, unlock the configuration, and release the resources (for example, memory) used by the session. After this operation, you exit the XML view.

XML example

# Terminate the current NETCONF session.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<close-session/>
</rpc>
```

edit-config: create

Usage guidelines

This operation creates target configuration items.

To use the **create** attribute in an <edit-config> operation, you must specify the target configuration item.

- If the table supports creating a target configuration item and the item does not exist, the operation creates the item and configures the item.
- If the specified item already exists, a data-exist error message is returned.

XML example

# Set the buffer size to 120.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
```

```
        <Syslog xmlns="http://www.nsfocus.com.cn/netconf/config:1.0"
xc:operation="create">
           <LogBuffer>
             <BufferSize>120</BufferSize>
          </LogBuffer>
        </Syslog>
      </top>
    </config>
  </edit-config>
</rpc>
```

edit-config: delete

Usage guidelines

This operation deletes the specified configuration.

- If the specified target has only the table index, the operation removes all configuration of the specified target, and the target itself.
- If the specified target has the table index and configuration data, the operation removes the specified configuration data of this target.
- If the specified target does not exist, an error message is returned, showing that the target does not exist.

XML example

The syntax is the same as the edit-config message with the **create** attribute. Change the operation attribute from **create** to **delete**.

edit-config: merge

Usage guidelines

This operation commits target configuration items to the running configuration.

To use the **merge** attribute in an <edit-config> operation, you must specify the target configuration item (on a specific level):

- If the specified item exists, the operation directly updates the setting for the item.
- If the specified item does not exist, the operation creates the item and configures the item.
- If the specified item does not exist and it cannot be created, an error message is returned.

XML example

The XML data format is the same as the edit-config message with the **create** attribute. Change the operation attribute from **create** to **merge**.

edit-config: remove

Usage guidelines

This operation removes the specified configuration.

- If the specified target has only the table index, the operation removes all configuration of the specified target, and the target itself.
- If the specified target has the table index and configuration data, the operation removes the specified configuration data of this target.

- If the specified target does not exist, or the XML message does not specify any targets, a success message is returned.

XML example

The syntax is the same as the edit-config message with the **create** attribute. Change the operation attribute from **create** to **remove**.

edit-config: replace

Usage guidelines

This operation replaces the specified configuration.

- If the specified target exists, the operation replaces the configuration of the target with the configuration carried in the message.
- If the specified target does not exist but is allowed to be created, the operation creates the target and then applies the configuration.
- If the specified target does not exist and is not allowed to be created, the operation is not conducted and an invalid-value error message is returned.

XML example

The syntax is the same as the edit-config message with the **create** attribute. Change the operation attribute from **create** to **replace**.

edit-config: test-option

Usage guidelines

This operation determines whether to commit a configuration item in an <edit-configure> operation. The <test-option> element has one of the following values:

- **test-then-set**—Performs a syntax check, and commits an item if the item passes the check. If the item fails the check, the item is not committed. This is the default test-option value.
- **set**—Commits the item without performing a syntax check.
- **test-only**—Performs only a syntax check. If the item passes the check, a success message is returned. Otherwise, an error message is returned.

XML example

# Test the configuration for an interface.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <test-option>test-only</test-option>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr xc:operation="merge">
          <Interfaces>
            <Interface>
              <IfIndex>262</IfIndex>
              <Description>222</Description>
                <ConfigSpeed>2</ConfigSpeed>
                <ConfigDuplex>1</ConfigDuplex>
               </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </config>
  </edit-config>
</rpc>
```

edit-config: default-operation

Usage guidelines

This operation modifies the running configuration of the device by using the default operation method.

NETCONF uses one of the following operation attributes to modify configuration: **merge**, **create**, **delete**, and **replace** If you do not specify an operation attribute for an edit-config message, NETCONF uses the default operation method. Your setting of the value for the <default-operation> element takes effect only once. If you do not specify an operation attribute or the default operation method for an <edit-config> message, **merge** always applies.

The <default-operation> element has the following values:

- **merge**—Default value for the <default-operation> element.
- **replace**—Value used when the operation attribute is not specified and the default operation method is specified as **replace**.
- **none**—Value used when the operation attribute is not specified and the default operation method is specified as **none**. If this value is specified, the <edit-config> operation is used only for schema verification rather than issuing a configuration. If the schema verification is passed, a successful message is returned. Otherwise, an error message is returned.

XML example

# Issue an empty operation for schema verification purposes.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <default-operation>none</default-operation>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr>
          <Interfaces>
            <Interface>
              <IfIndex>262</IfIndex>
              <Description>222222</Description>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </config>
  </edit-config>
</rpc>
```

edit-config: error-option

Usage guidelines

This operation determines the action to take in case of a configuration error.

The <error-option> element has the following values:

- **stop-on-error**—Stops the operation and returns an error message. This is the default error-option value.
- **continue-on-error**—Continues the operation and returns an error message.
- **rollback-on-error**—Rolls back the configuration.

XML example

# Issue the configuration for two interfaces with the <error-option> element value as **continue-on-error**.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>   <error-option>continue-on-error</error-option>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr xc:operation="merge">
          <Interfaces>
            <Interface>
              <IfIndex>262</IfIndex>
              <Description>222</Description>
                <ConfigSpeed>1024</ConfigSpeed>
                <ConfigDuplex>1</ConfigDuplex>
            </Interface>
            <Interface>
              <IfIndex>263</IfIndex>
              <Description>333</Description>
                <ConfigSpeed>1024</ConfigSpeed>
                <ConfigDuplex>1</ConfigDuplex>
            </Interface>
          </Interfaces>
        </Ifmgr>
      </top>
    </config>
  </edit-config>
</rpc>
```

edit-config: incremental

Usage guidelines

This operation adds configuration data to a column without affecting the original data.

The **incremental** attribute applies to a list column such as the vlan permitlist column.

You can use the **incremental** attribute for <edit-config> operations except the <replace> operation.

Support for the **incremental** attribute varies by module. For more information, see NETCONF XML API documents.

XML example

# Add VLANs 1 through 10 to an untagged VLAN list that has untagged VLANs 12 through 15.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nsfocus="http://www.nsfocus.com.cn/netconf/base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <VLAN xc:operation="merge">
          <HybridInterfaces>
            <Interface>
              <IfIndex>262</IfIndex>
              <UntaggedVlanList  nsfocus:incremental="true">1-10</UntaggedVlanList>
               </Interface>
          </HybridInterfaces>
        </VLAN>
      </top>
    </config>
  </edit-config>
</rpc>
```

get

Usage guidelines

This operation retrieves device configuration and state information.

XML example

# Retrieve device configuration and state information for the Syslog module.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xc="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Syslog>
        </Syslog>
      </top>
    </filter>
  </get>
</rpc>
```

get-bulk

Usage guidelines

This operation retrieves a number of data entries (including device configuration and state information) starting from the data entry next to the one with the specified index.

XML example

# Retrieve device configuration and state information for all interfaces.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-bulk>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/data:1.0">
        <Ifmgr>
          <Interfaces xc:count="5"
xmlns:xc="http://www.nsfocus.com.cn/netconf/base:1.0">
            <Interface/>
          </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get-bulk>
</rpc>
```

get-bulk-config

Usage guidelines

This operation retrieves a number of non-default configuration data entries starting from the data entry next to the one with the specified index.

XML example

# Retrieve non-default configuration for all interfaces.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-bulk-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr>
        </Ifmgr>
      </top>
    </filter>
  </get-bulk-config>
</rpc>
```

get-config

Usage guidelines

This operation retrieves non-default configuration data. If no non-default configuration data exists, the device returns a response with empty data.

XML example

# Retrieve non-default configuration data for the interface table.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:xc="http://www.nsfocus.com.cn/netconf/base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <top xmlns="http://www.nsfocus.com.cn/netconf/config:1.0">
        <Ifmgr>
           <Interfaces>
              <Interface/>
            </Interfaces>
        </Ifmgr>
      </top>
    </filter>
  </get-config>
</rpc>
```

get-sessions

Usage guidelines

This operation retrieves information about all NETCONF sessions in the system. You cannot specify a session ID to retrieve information about a specific NETCONF session.

XML example

# Retrieve information about all NETCONF sessions in the system.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get-sessions/>
</rpc>
```

kill-session

Usage guidelines

This operation terminates the NETCONF session for another user. This operation cannot terminate the NETCONF session for the current user.

XML example

# Terminate the NETCONF session with session ID 1.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <kill-session>
    <session-id>1</session-id>
  </kill-session>
</rpc>
```

load

Usage guidelines

This operation loads the configuration. After the device finishes a <load> operation, the configuration in the specified file is merged into the running configuration of the device.

XML example

# Merge the configuration in file **a1.cfg** to the running configuration of the device.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <load>
    <file>a1.cfg</file>
  </load>
</rpc>
```

lock

Usage guidelines

This operation locks the configuration. After the configuration is locked, you cannot perform <edit-config> operations. Other operations are allowed.

After a user locks the configuration, other users cannot use NETCONF or any other configuration methods such as CLI and SNMP to configure the device.

XML example

# Lock the configuration.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <lock>
    <target>
        <running/>
    </target>
</lock>
</rpc>
```

rollback

Usage guidelines

This operation rolls back the configuration. To do so, you must specify the configuration file in the <file> element. After the device finishes the <rollback> operation, the current device configuration is totally replaced with the configuration in the specified configuration file.

XML example

# Roll back the running configuration to the configuration in file **1A.cfg**.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rollback>
    <file>1A.cfg</file>
  </rollback>
</rpc>
```

save

Usage guidelines

This operation saves the running configuration. You can use the <file> element to specify a file for saving the configuration. If the text does not include the file column, the running configuration is automatically saved to the main next-startup configuration file.

The **OverWrite** attribute determines whether the running configuration overwrites the original configuration file when the specified file already exists.

The **Binary-only** attribute determines whether to save the running configuration only to the binary configuration file.

XML example

# Save the running configuration to file **test.cfg**.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <save OverWrite="false" Binary-only="true">
    <file>test.cfg</file>
  </save>
</rpc>
```

unlock

Usage guidelines

This operation unlocks the configuration, so other users can configure the device.

Terminating a NETCONF session automatically unlocks the configuration.

XML example

# Unlock the configuration.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<unlock>
    <target>
        <running/>
    </target>
</unlock>
</rpc>
```

Contents

Configuring CWMP

About CWMP

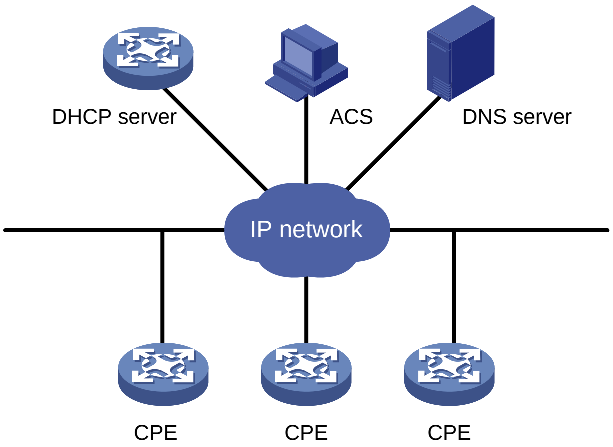
CPE WAN Management Protocol (CWMP), also called "TR-069," is a DSL Forum technical specification for remote management of network devices.

The protocol was initially designed to provide remote autoconfiguration through a server for large numbers of dispersed end-user devices in a network. CWMP can be used on different types of networks, including Ethernet.

CWMP network framework

Figure 1 shows a basic CWMP network framework.

Figure 1 CWMP network framework



A basic CWMP network includes the following network elements:

- **ACS**—Autoconfiguration server, the management device in the network.
- **CPE**—Customer premises equipment, the managed device in the network.
- **DNS server**—Domain name system server. CWMP defines that the ACS and the CPE use URLs to identify and access each other. DNS is used to resolve the URLs.
- **DHCP server**—Assigns ACS attributes along with IP addresses to CPEs when the CPEs are powered on. DHCP server is optional in CWMP. With a DHCP server, you do not need to configure ACS attributes manually on each CPE. The CPEs can contact the ACS automatically when they are powered on for the first time.

The device is operating as a CPE in the CWMP framework.

Basic CWMP functions

You can autoconfigure and upgrade CPEs in bulk from the ACS.

Autoconfiguration

You can create configuration files for different categories of CPEs on the ACS. Based on the device models and serial numbers of the CPEs, the ACS verifies the categories of the CPEs and issues the associated configuration to them.

The following are methods available for the ACS to issue configuration to the CPE:

- Transfers the configuration file to the CPE, and specifies the file as the next-startup configuration file. At a reboot, the CPE starts up with the ACS-specified configuration file.
- Runs the configuration in the CPE's RAM. The configuration takes effect immediately on the CPE. For the running configuration to survive a reboot, you must save the configuration on the CPE.

CPE software management

The ACS can manage CPE software upgrade.

When the ACS finds a software version update, the ACS notifies the CPE to download the software image file from a specific location. The location can be the URL of the ACS or an independent file server.

If the CPE successfully downloads the software image file and the file is validated, the CPE notifies the ACS of a successful download. If the CPE fails to download the software image file or the file is invalidated, the CPE notifies the ACS of an unsuccessful download.

Data backup

The ACS can require the CPE to upload a configuration file or log file to a specific location. The destination location can be the ACS or a file server.

CPE status and performance monitoring

The ACS can monitor the status and performance of CPEs. Table 1 shows the available CPE status and performance objects for the ACS to monitor.

Table 1 CPE status and performance objects available for the ACS to monitor

Category	Objects	Remarks
Device information	Manufacturer ManufacturerOUI SerialNumber HardwareVersion SoftwareVersion	N/A
Operating status and information	DeviceStatus UpTime	N/A
Configuration file	ConfigFile	Local configuration file stored on CPE for upgrade. The ACS can issue configuration to the CPE by transferring a configuration file to the CPE or running the configuration in CPE's RAM.
CWMP settings	ACS URL	URL address of the ACS to which the CPE initiates a CWMP connection. This object is also used for main/backup ACS switchover.
	ACS username ACS password	When the username and password of the ACS are changed, the ACS changes the ACS username and password on the CPE to the new username and password. When a main/backup ACS switchover occurs, the main ACS also changes the ACS username and password to the backup ACS username and password.
	PeriodicInformEnable	Whether to enable or disable the periodic Inform feature.
	PeriodicInformInterval	Interval for periodic connection from the CPE

Category	Objects	Remarks
		to the ACS for configuration and software update.
	PeriodicInformTime	Scheduled time for connection from the CPE to the ACS for configuration and software update.
	ConnectionRequestURL (CPE URL)	N/A
	ConnectionRequestUsername (CPE username) ConnectionRequestPassword (CPE password)	CPE username and password for authentication from the ACS to the CPE.

How CWMP works

RPC methods

CWMP uses remote procedure call (RPC) methods for bidirectional communication between CPE and ACS. The RPC methods are encapsulated in HTTP or HTTPS.

Table 2 shows the primary RPC methods used in CWMP.

Table 2 RPC methods

RPC method	Description
Get	The ACS obtains the values of parameters on the CPE.
Set	The ACS modifies the values of parameters on the CPE.
Inform	The CPE sends an Inform message to the ACS for the following purposes: • Initiates a connection to the ACS. • Reports configuration changes to the ACS. • Periodically updates CPE settings to the ACS.
Download	The ACS requires the CPE to download a configuration or software image file from a specific URL for software or configuration update.
Upload	The ACS requires the CPE to upload a file to a specific URL.
Reboot	The ACS reboots the CPE remotely for the CPE to complete an upgrade or recover from an error condition.

Autoconnect between ACS and CPE

The CPE automatically initiates a connection to the ACS when one of the following events occurs:

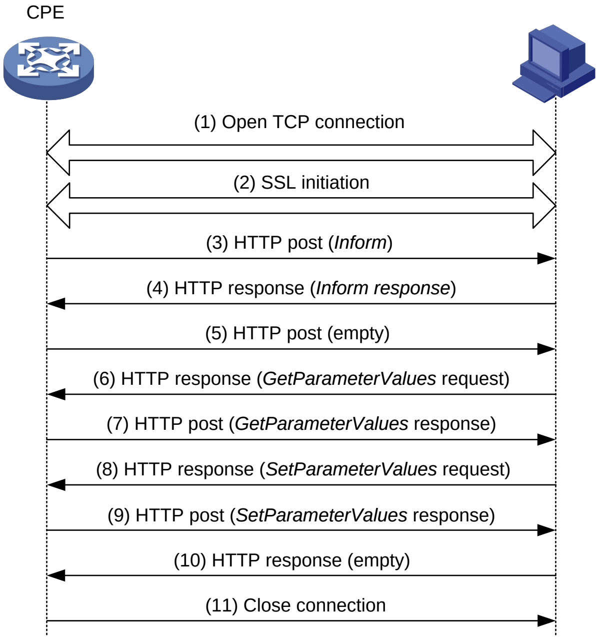
- ACS URL change. The CPE initiates a connection request to the new ACS URL.
- CPE startup. The CPE initiates a connection to the ACS after the startup.
- Timeout of the periodic Inform interval. The CPE re-initiates a connection to the ACS at the Inform interval.
- Expiration of the scheduled connection initiation time. The CPE initiates a connection to the ACS at the scheduled time.

CWMP connection establishment

Step 1 through step 5 in Figure 2 show the procedure of establishing a connection between the CPE and the ACS.

1. After obtaining the basic ACS parameters, the CPE initiates a TCP connection to the ACS.
2. If HTTPS is used, the CPE and the ACS initialize SSL for a secure HTTP connection.
3. The CPE sends an Inform message in HTTPS to initiate a CWMP session.
4. After the CPE passes authentication, the ACS returns an Inform response to establish the session.
5. After sending all requests, the CPE sends an empty HTTP post message.

Figure 2 CWMP connection establishment



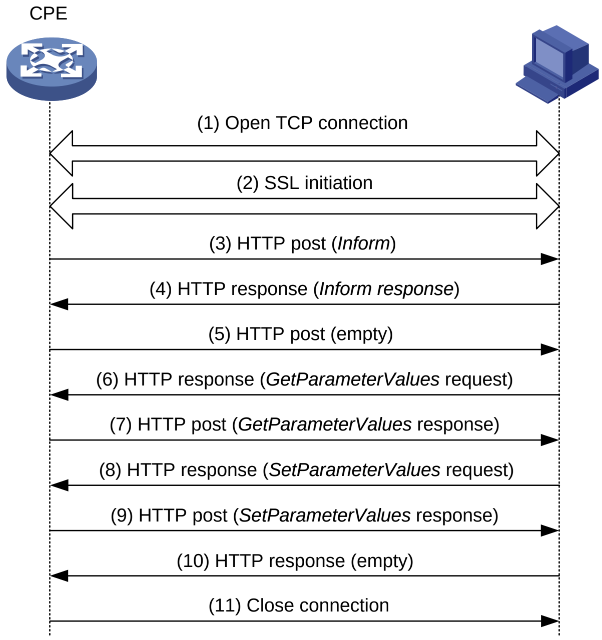
CPE
(1) Open TCP connection
(2) SSL initiation
(3) HTTP post (*Inform*)
(4) HTTP response (*Inform response*)
(5) HTTP post (empty)
(6) HTTP response (*GetParameterValues* request)
(7) HTTP post (*GetParameterValues* response)
(8) HTTP response (*SetParameterValues* request)
(9) HTTP post (*SetParameterValues* response)
(10) HTTP response (empty)
(11) Close connection

Main/backup ACS switchover

Typically, two ACSs are used in a CWMP network for consecutive monitoring on CPEs. When the main ACS needs to reboot, it points the CPE to the backup ACS. Step 6 through step 11 in Figure 3 show the procedure of a main/backup ACS switchover.

1. Before the main ACS reboots, it queries the ACS URL set on the CPE.
2. The CPE replies with its ACS URL setting.
3. The main ACS sends a Set request to change the ACS URL on the CPE to the backup ACS URL.
4. After the ACS URL is modified, the CPE sends a response.
5. The main ACS sends an empty HTTP message to notify the CPE that it has no other requests.
6. The CPE closes the connection, and then initiates a new connection to the backup ACS URL.

Figure 3 Main and backup ACS switchover


CPE
(1) Open TCP connection
(2) SSL initiation
(3) HTTP post (*Inform*)
(4) HTTP response (*Inform response*)
(5) HTTP post (empty)
(6) HTTP response (*GetParameterValues* request)
(7) HTTP post (*GetParameterValues* response)
(8) HTTP response (*SetParameterValues* request)
(9) HTTP post (*SetParameterValues* response)
(10) HTTP response (empty)
(11) Close connection

Restrictions and guidelines: CWMP configuration

You can configure ACS and CPE attributes from the CPE's CLI, the DHCP server, or the ACS. For an attribute, the CLI- and ACS-assigned values have higher priority than the DHCP-assigned value. The CLI- and ACS-assigned values overwrite each other, whichever is assigned later.

This document only describes configuring ACS and CPE attributes from the CLI and DHCP server. For more information about configuring and using the ACS, see ACS documentation.

CWMP tasks at a glance

To configure CWMP, perform the following tasks:

1. Enabling CWMP from the CLI

   You can also enable CWMP from a DHCP server.
2. Configuring ACS attributes
   a. Configuring the preferred ACS attributes
   b. (Optional.) Configuring the default ACS attributes from the CLI
3. Configuring CPE attributes
   a. Specifying an SSL client policy for HTTPS connection to ACS

      This task is required when the ACS uses HTTPS for secure access.
   b. (Optional.) Configuring ACS authentication parameters
   c. (Optional.) Configuring the provision code
   d. (Optional.) Configuring the CWMP connection interface
   e. (Optional.) Configuring autoconnect parameters
   f. (Optional.) Setting the close-wait timer
   g. (Optional.) Enabling NAT traversal for the CPE

Enabling CWMP from the CLI

4. Enter system view.

   system-view

5. Enter CWMP view.

   cwmp

6. Enable CWMP.

   cwmp enable

   By default, CWMP is disabled.

Configuring ACS attributes

About ACS attributes

You can configure two sets of ACS attributes for the CPE: preferred and default.

- The preferred ACS attributes are configurable from the CPE's CLI, the DHCP server, and ACS.
- The default ACS attributes are configurable only from the CLI.

If the preferred ACS attributes are not configured, the CPE uses the default ACS attributes for connection establishment.

Configuring the preferred ACS attributes

Assigning ACS attributes from the DHCP server

The DHCP server in a CWMP network assigns the following information to CPEs:

- IP addresses for the CPEs.
- DNS server address.
- ACS URL and ACS login authentication information.

This section introduces how to use DHCP option 43 to assign the ACS URL and ACS login authentication username and password. For more information about DHCP and DNS, see *Layer 3—IP Services Configuration Guide*.

If the DHCP server is an NSFOCUS device, you can configure DHCP option 43 by using the **option 43 hex 01***length URL username password* command.

- *length*—A hexadecimal number that indicates the total length of the *length*, *URL*, *username*, and *password* arguments, including the spaces between these arguments. No space is allowed between the **01** keyword and the length value.
- *URL*—ACS URL.
- *username*—Username for the CPE to authenticate to the ACS.
- *password*—Password for the CPE to authenticate to the ACS.

NOTE:

The ACS URL, username and password must use the hexadecimal format and be space separated.

The following example configures the ACS address as **http://169.254.76.31:7547**, username as **1234**, and password as **5678**:

```
<Sysname> system-view
```

```
[Sysname] dhcp server ip-pool 0
[Sysname-dhcp-pool-0] option 43 hex
0123687474703A2F2F3136392E3235342E37362E33313A3735343720313233342035363738
```

Table 3 Hexadecimal forms of the ACS attributes

Attribute	Attribute value	Hexadecimal form
Length	35 characters	23
ACS URL	http://169.254.76.31:7547	687474703A2F2F3136392E3235342E37362E33313A3735343720 **NOTE:** The two ending digits (20) represent the space.
ACS connect username	1234	3132333420 **NOTE:** The two ending digits (20) represent the space.
ACS connect password	5678	35363738

Configuring the preferred ACS attributes from the CLI

1. Enter system view.

   `system-view`

2. Enter CWMP view.

   `cwmp`

3. Configure the preferred ACS URL.

   `cwmp acs url` *`url`*

   By default, no preferred ACS URL has been configured.

4. Configure the username for authentication to the preferred ACS URL.

   `cwmp acs username` *`username`*

   By default, no username has been configured for authentication to the preferred ACS URL.

5. (Optional.) Configure the password for authentication to the preferred ACS URL.

   `cwmp acs password` { **`cipher`** | **`simple`** } *`string`*

   By default, no password has been configured for authentication to the preferred ACS URL.

Configuring the default ACS attributes from the CLI

1. Enter system view.

   `system-view`

2. Enter CWMP view.

   `cwmp`

3. Configure the default ACS URL.

   `cwmp acs default url` *`url`*

   By default, no default ACS URL has been configured.

4. Configure the username for authentication to the default ACS URL.

   `cwmp acs default username` *`username`*

   By default, no username has been configured for authentication to the default ACS URL.

5. (Optional.) Configure the password for authentication to the default ACS URL.

cwmp acs default password { **cipher** | **simple** } *string*

By default, no password has been configured for authentication to the default ACS URL.

Configuring CPE attributes

About CPE attributes

You can configure the following CPE attributes only from the CPE's CLI.

- CWMP connection interface.
- NAT traversal.
- Maximum number of connection retries.
- SSL client policy for HTTPS connection to ACS.

For other CPE attribute values, you can assign them to the CPE from the CPE's CLI or the ACS. The CLI- and ACS-assigned values overwrite each other, whichever is assigned later.

Specifying an SSL client policy for HTTPS connection to ACS

About this task

This task is required when the ACS uses HTTPS for secure access. CWMP uses HTTP or HTTPS for data transmission. When HTTPS is used, the ACS URL begins with **https://**. You must specify an SSL client policy for the CPE to authenticate the ACS for HTTPS connection establishment.

Restrictions and guidelines

Versions B64D028 and higher use upgraded SSL cipher suites and require the DH parameters to contain a minimum of 1024 bits. If the ACS still uses the old version, SSL connections might fail to be established and CPEs might fail to connect to the ACS.

To solve this issue, upgrade the ACS or execute the **prefer-cipher** command on CPEs to specify the SSL client policy to prefer an non-DHE RSA algorithm. For more information about the **prefer-cipher** command, see *Security Command Reference*.

Prerequisites

Before you perform this task, first create an SSL client policy. For more information about configuring SSL client policies, see *Security Configuration Guide*.

Procedure

1. Enter system view.
   system-view
2. Enter CWMP view.
   cwmp
3. Specify an SSL client policy.
   ssl client-policy *policy-name*
   By default, no SSL client policy is specified.

Configuring ACS authentication parameters

About this task

To protect the CPE against unauthorized access, configure a CPE username and password for ACS authentication. When an ACS initiates a connection to the CPE, the ACS must provide the correct username and password.

Procedure

1. Enter system view.

   system-view

2. Enter CWMP view.

   cwmp

3. Configure the username for authentication to the CPE.

   cwmp cpe username *username*

   By default, no username has been configured for authentication to the CPE.

4. (Optional.) Configure the password for authentication to the CPE.

   cwmp cpe password { **cipher** | **simple** } *string*

   By default, no password has been configured for authentication to the CPE.

   The password setting is optional. You can specify only a username for authentication.

Configuring the provision code

About this task

The ACS can use the provision code to identify services assigned to each CPE. For correct configuration deployment, make sure the same provision code is configured on the CPE and the ACS. For information about the support of your ACS for provision codes, see the ACS documentation.

Procedure

1. Enter system view.

   system-view

2. Enter CWMP view.

   cwmp

3. Configure the provision code.

   cwmp cpe provision-code *provision-code*

   The default provision code is **PROVISIONINGCODE**.

Configuring the CWMP connection interface

About this task

The CWMP connection interface is the interface that the CPE uses to communicate with the ACS. To establish a CWMP connection, the CPE sends the IP address of this interface in the Inform messages, and the ACS replies to this IP address.

Typically, the CPE selects the CWMP connection interface automatically. If the CWMP connection interface is not the interface that connects the CPE to the ACS, the CPE fails to establish a CWMP connection with the ACS. In this case, you need to manually set the CWMP connection interface.

Procedure

1. Enter system view.

   system-view

2. Enter CWMP view.

   cwmp

3. Specify the interface that connects to the ACS as the CWMP connection interface.

   cwmp cpe connect interface *interface-type interface-number*

   By default, no CWMP connection interface is specified.

Configuring autoconnect parameters

About this task

You can configure the CPE to connect to the ACS periodically, or at a scheduled time for configuration or software update.

The CPE retries a connection automatically when one of the following events occurs:

- The CPE fails to connect to the ACS. The CPE considers a connection attempt as having failed when the close-wait timer expires. This timer starts when the CPE sends an Inform request. If the CPE fails to receive a response before the timer expires, the CPE resends the Inform request.
- The connection is disconnected before the session on the connection is completed.

To protect system resources, limit the number of retries that the CPE can make to connect to the ACS.

Configuring the periodic Inform feature

1. Enter system view.

   system-view

2. Enter CWMP view.

   cwmp

3. Enable the periodic Inform feature.

   cwmp cpe inform interval enable

   By default, this function is disabled.

4. Set the Inform interval.

   cwmp cpe inform interval *interval*

   By default, the CPE sends an Inform message to start a session every 600 seconds.

Scheduling a connection initiation

1. Enter system view.

   system-view

2. Enter CWMP view.

   cwmp

3. Schedule a connection initiation.

   cwmp cpe inform time *time*

   By default, no connection initiation has been scheduled.

Setting the maximum number of connection retries

1. Enter system view.

```
system-view
```

2. Enter CWMP view.

```
cwmp
```

3. Set the maximum number of connection retries.

```
cwmp cpe connect retry retries
```

By default, the CPE retries a failed connection until the connection is established.

Setting the close-wait timer

About this task

The close-wait timer specifies the following:

- The maximum amount of time the CPE waits for the response to a session request. The CPE determines that its session attempt has failed when the timer expires.
- The amount of time the connection to the ACS can be idle before it is terminated. The CPE terminates the connection to the ACS if no traffic is sent or received before the timer expires.

Procedure

1. Enter system view.

```
system-view
```

2. Enter CWMP view.

```
cwmp
```

3. Set the close-wait timer.

```
cwmp cpe wait timeout seconds
```

By default, the close-wait timer is 30 seconds.

Enabling NAT traversal for the CPE

About this task

For the connection request initiated from the ACS to reach the CPE, you must enable NAT traversal on the CPE when a NAT gateway resides between the CPE and the ACS.

The NAT traversal feature complies with RFC 3489 Simple Traversal of UDP Through NATs (STUN). The feature enables the CPE to discover the NAT gateway, and obtain an open NAT binding (a public IP address and port binding) through which the ACS can send unsolicited packets. The CPE sends the binding to the ACS when it initiates a connection to the ACS. For the connection requests sent by the ACS at any time to reach the CPE, the CPE maintains the open NAT binding. For more information about NAT, see *NAT Configuration Guide*.

Procedure

1. Enter system view.

```
system-view
```

2. Enter CWMP view.

```
cwmp
```

3. Enable NAT traversal.

```
cwmp cpe stun enable
```

By default, NAT traversal is disabled on the CPE.

Display and maintenance commands for CWMP

Execute **display** commands in any view.

Task	Command
Display CWMP configuration.	**display cwmp configuration**
Display the current status of CWMP.	**display cwmp status**

CWMP configuration examples

Example: Configuring CWMP

Network configuration

As shown in Figure 4, use IMC BIMS as the ACS to bulk-configure the devices (CPEs), and assign ACS attributes to the CPEs from the DHCP server.

The configuration files for the devices in equipment rooms A and B are **configure1.cfg** and **configure2.cfg**, respectively.

Figure 4 Network diagram

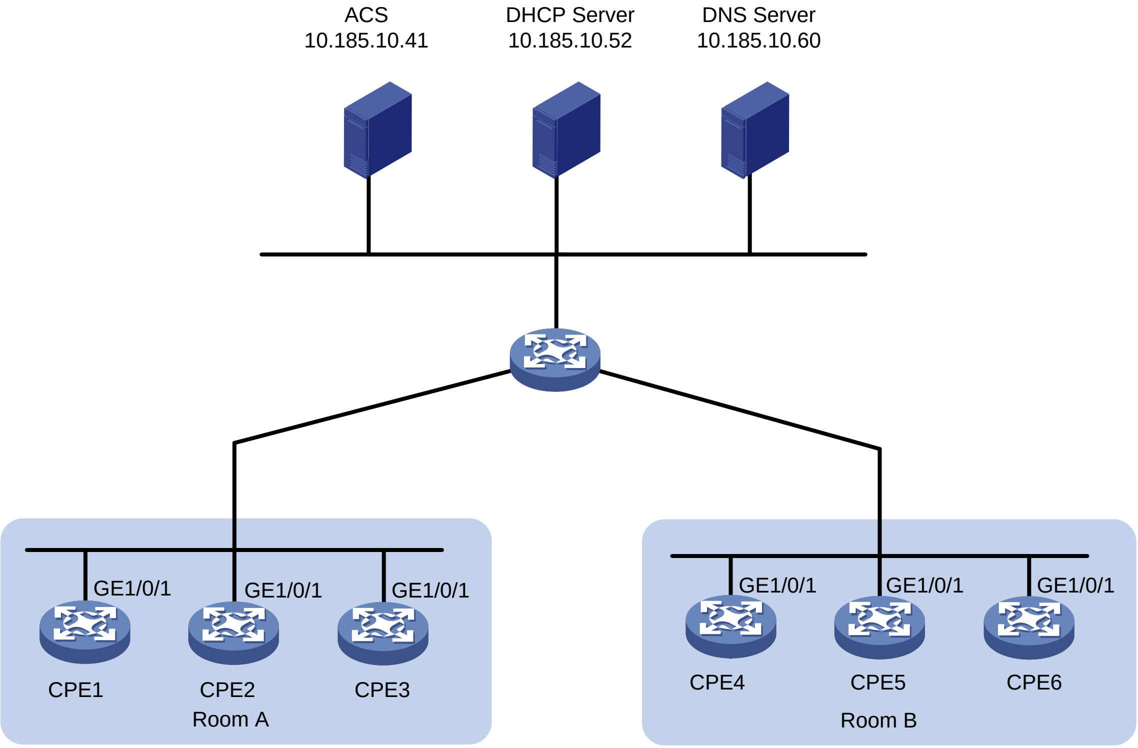


Table 4 shows the ACS attributes for the CPEs to connect to the ACS.

Table 4 ACS attributes

Item	Setting
Preferred ACS URL	http://10.185.10.41:9090

Item	Setting
ACS username	admin
ACS password	12345

Table 5 lists serial numbers of the CPEs.

Table 5 CPE list

Room	Device	Serial number
A	CPE1	210231A95YH10C000045
	CPE2	210235AOLNH12000010
	CPE3	210235AOLNH12000015
B	CPE4	210235AOLNH12000017
	CPE5	210235AOLNH12000020
	CPE6	210235AOLNH12000022

Configuring the ACS

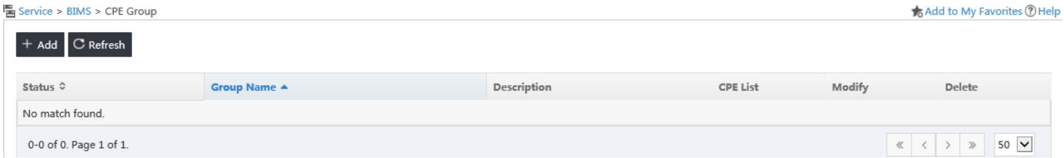
Figures in this section are for illustration only.

To configure the ACS:

1. Log in to the ACS:
   a. Launch a Web browser on the ACS configuration terminal.
   b. In the address bar of the Web browser, enter the ACS URL and port number. This example uses **http://10.185.10.41:8080/imc**.
   c. On the login page, enter the ACS login username and password, and then click **Login**.
2. Create a CPE group for each equipment room:
   a. Select **Service** > **BIMS** > **CPE Group** from the top navigation bar.

      The **CPE Group** page appears.

      Figure 5 CPE Group page

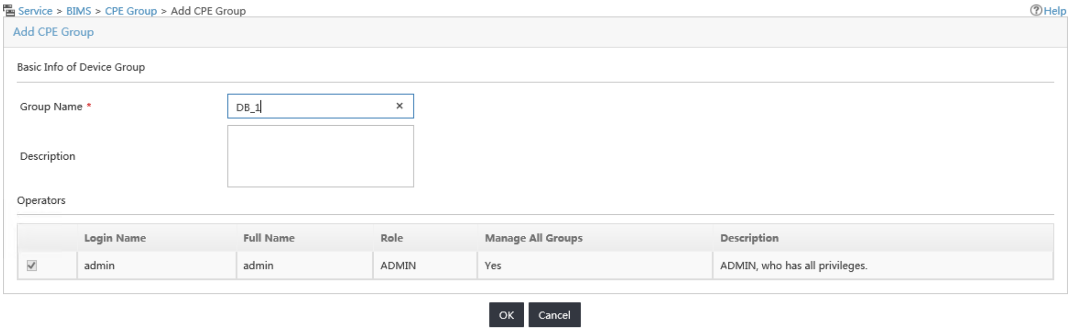


Status	Group Name	Description	CPE List	Modify	Delete
No match found.					

0-0 of 0. Page 1 of 1.

   b. Click **Add**.
   c. Enter a username, and then click **OK**.

      Figure 6 Adding a CPE group



Basic Info of Device Group

Group Name * DB_1

Description

Operators

	Login Name	Full Name	Role	Manage All Groups	Description
☑	admin	admin	ADMIN	Yes	ADMIN, who has all privileges.

d. Repeat the previous two steps to create a CPE group for CPEs in Room B.

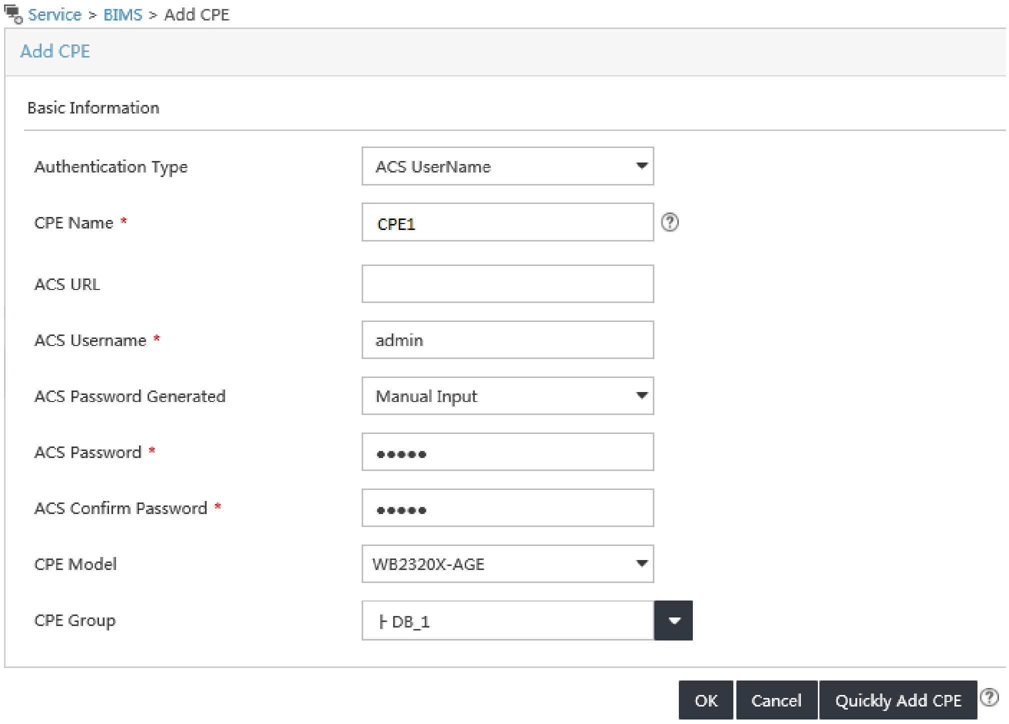
3. Add CPEs to the CPE group for each equipment room:

a. Select **Service** > **BIMS** > **Resource Management** > **Add CPE** from the top navigation bar.

b. On the **Add CPE** page, configure the following parameters:

- **Authentication Type**—Select **ACS UserName**.
- **CPE Name**—Enter a CPE name.
- **ACS Username**—Enter **admin**.
- **ACS Password Generated**—Select **Manual Input**.
- **ACS Password**—Enter a password for ACS authentication.
- **ACS Confirm Password**—Re-enter the password.
- **CPE Model**—Select the CPE model.
- **CPE Group**—Select the CPE group.

Figure 7 Adding a CPE



Service > BIMS > Add CPE

Add CPE

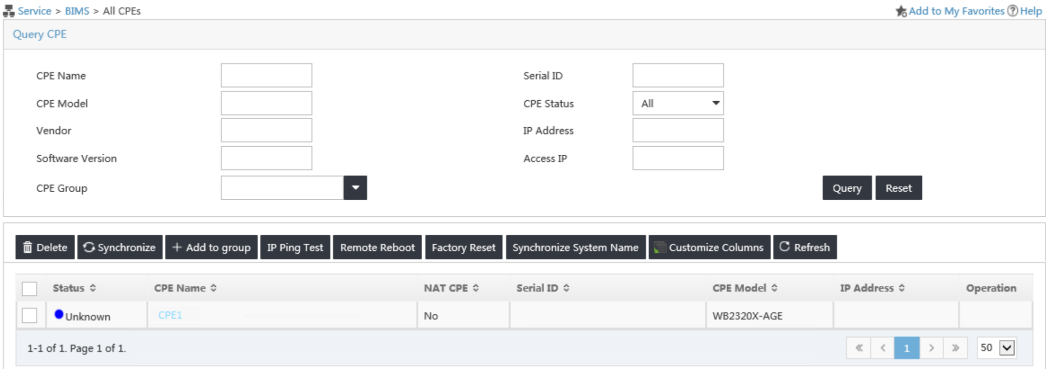
Basic Information

Authentication Type	ACS UserName
CPE Name *	CPE1
ACS URL	
ACS Username *	admin
ACS Password Generated	Manual Input
ACS Password *	●●●●●
ACS Confirm Password *	●●●●●
CPE Model	WB2320X-AGE
CPE Group	├ DB_1

c. Click **OK**.

d. Verify that the CPE has been added successfully from the **All CPEs** page.

Figure 8 Viewing CPEs



Service > BIMS > All CPEs

Add to My Favorites ⓘHelp

Query CPE

CPE Name		Serial ID	
CPE Model		CPE Status	All
Vendor		IP Address	
Software Version		Access IP	
CPE Group			

□	Status	CPE Name	NAT CPE	Serial ID	CPE Model	IP Address	Operation
□	● Unknown	CPE1	No		WB2320X-AGE		

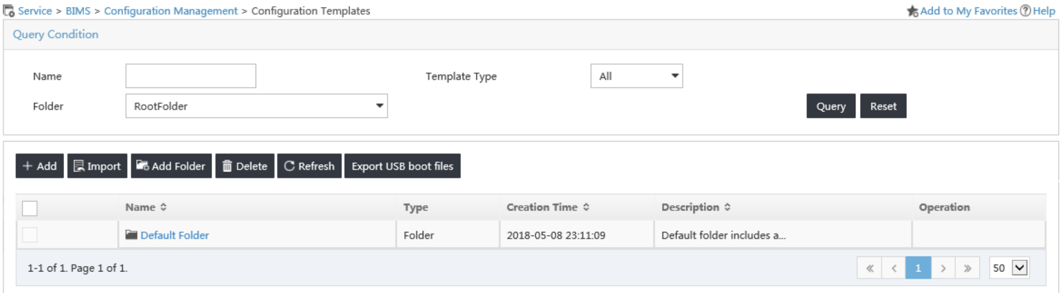
1-1 of 1. Page 1 of 1.

e. Repeat the previous steps to add CPE 2 and CPE 3 to the CPE group for Room A, and add CPEs in Room B to the CPE group for Room B.

4. Configure a configuration template for each equipment room:

a. Select **Service** > **BIMS** > **Configuration Management** > **Configuration Templates** from the top navigation bar.

Figure 9 Configuration Templates page



Service > BIMS > Configuration Management > Configuration Templates

Add to My Favorites ⓘHelp

Query Condition

Name		Template Type	All
Folder	RootFolder		

□	Name	Type	Creation Time	Description	Operation
□	Default Folder	Folder	2018-05-08 23:11:09	Default folder includes a...	

1-1 of 1. Page 1 of 1.

b. Click **Import**.

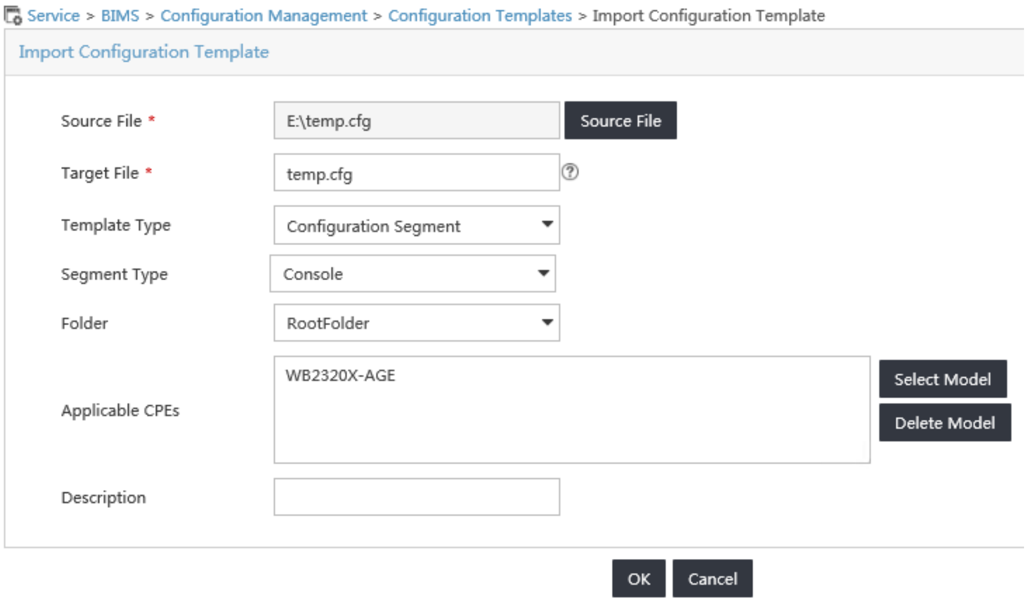
c. Select a source configuration file, select **Configuration Segment** as the template type, and then click **OK**.

The created configuration template will be displayed in the **Configuration Template** list after a successful file import.

IMPORTANT:

If the first command in the configuration template file is **`system-view`**, make sure no characters exist in front of the command.

Figure 10 Importing a configuration template

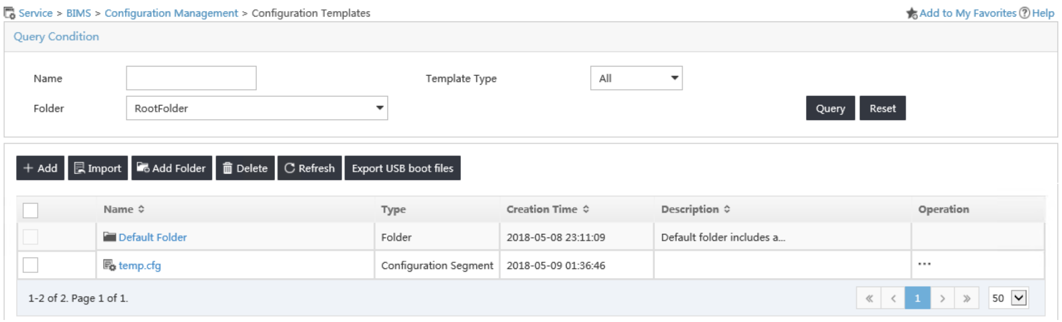


Service > BIMS > Configuration Management > Configuration Templates > Import Configuration Template

Import Configuration Template

Source File *	E:\temp.cfg [Source File]
Target File *	temp.cfg
Template Type	Configuration Segment
Segment Type	Console
Folder	RootFolder
Applicable CPEs	WB2320X-AGE [Select Model] [Delete Model]
Description	

Figure 11 Configuration Template list



Service > BIMS > Configuration Management > Configuration Templates

Name		Template Type	All
Folder	RootFolder		

□	Name	Type	Creation Time	Description	Operation
□	Default Folder	Folder	2018-05-08 23:11:09	Default folder includes a...	
□	temp.cfg	Configuration Segment	2018-05-09 01:36:46		···

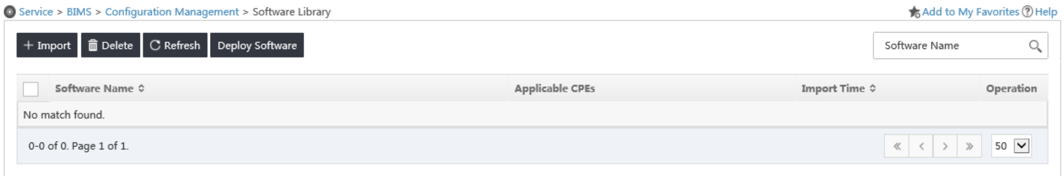
1-2 of 2. Page 1 of 1.

d. Repeat the previous steps to configure a configuration template for Room B.

5. Add software library entries:

a. Select **Service** > **BIMS** > **Configuration Management** > **Software Library** from the top navigation bar.

Figure 12 Software Library page



Service > BIMS > Configuration Management > Software Library

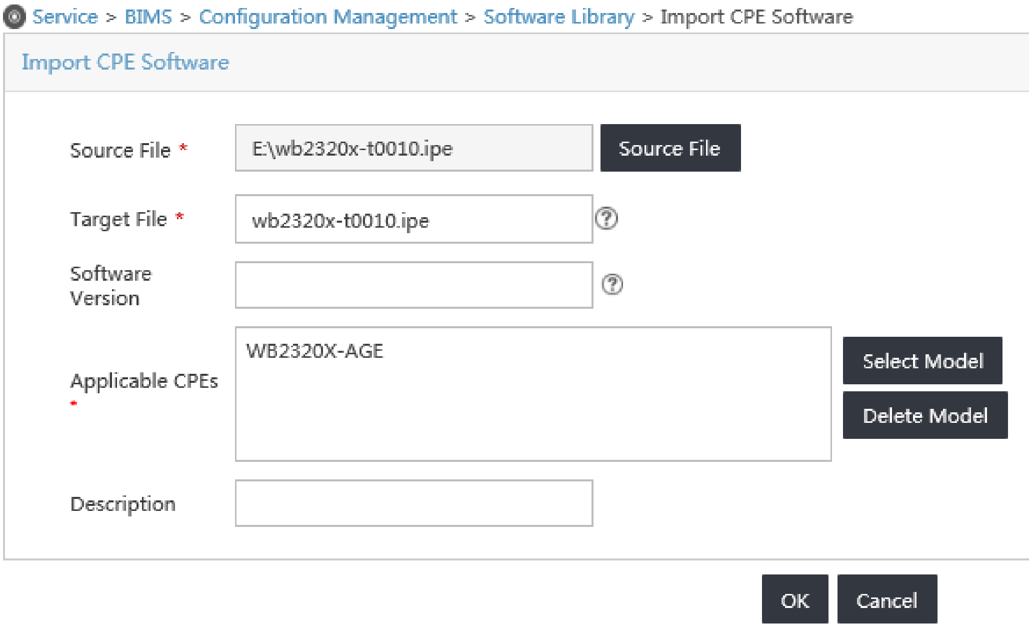
□	Software Name	Applicable CPEs	Import Time	Operation
No match found.				

0-0 of 0. Page 1 of 1.

b. Click **Import**.

c. Select a source file, and then click **OK**.

Figure 13 Importing CPE software



Service > BIMS > Configuration Management > Software Library > Import CPE Software

Import CPE Software

Source File *: E:\wb2320x-t0010.ipe

Target File *: wb2320x-t0010.ipe

Software Version:

Applicable CPEs *: WB2320X-AGE

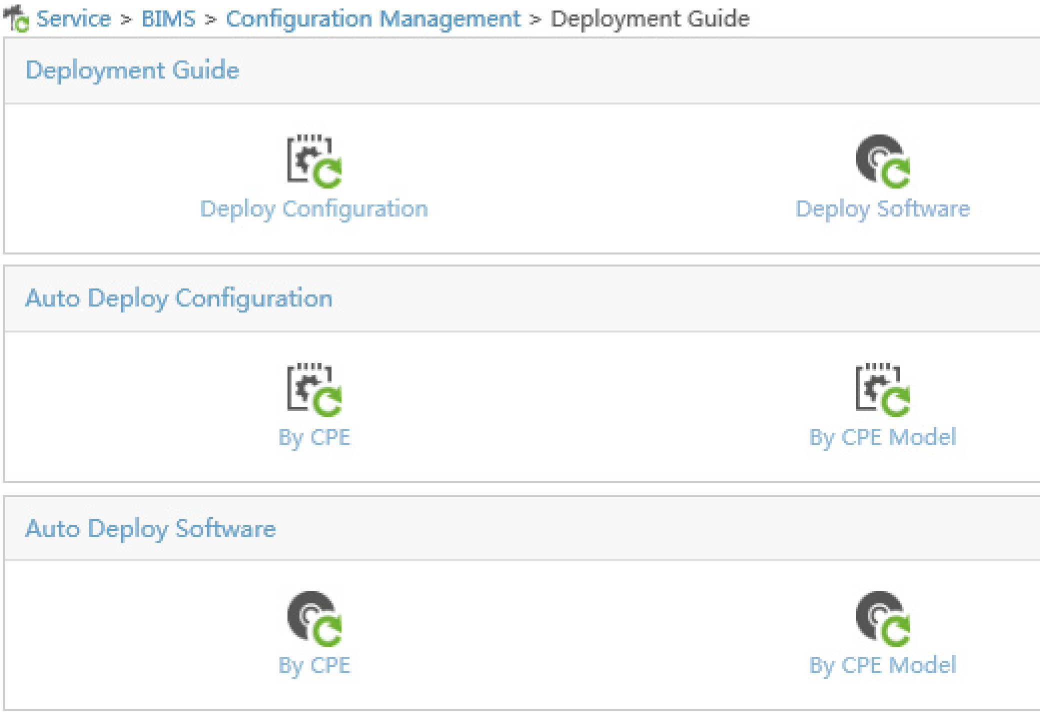
Description:

d. Repeat the previous steps to add software library entries for CPEs of different models.

6. Create an auto-deployment task for each equipment room:

a. Select **Service** > **BIMS** > **Configuration Management** > **Deployment Guide** from the top navigation bar.

Figure 14 Deployment Guide



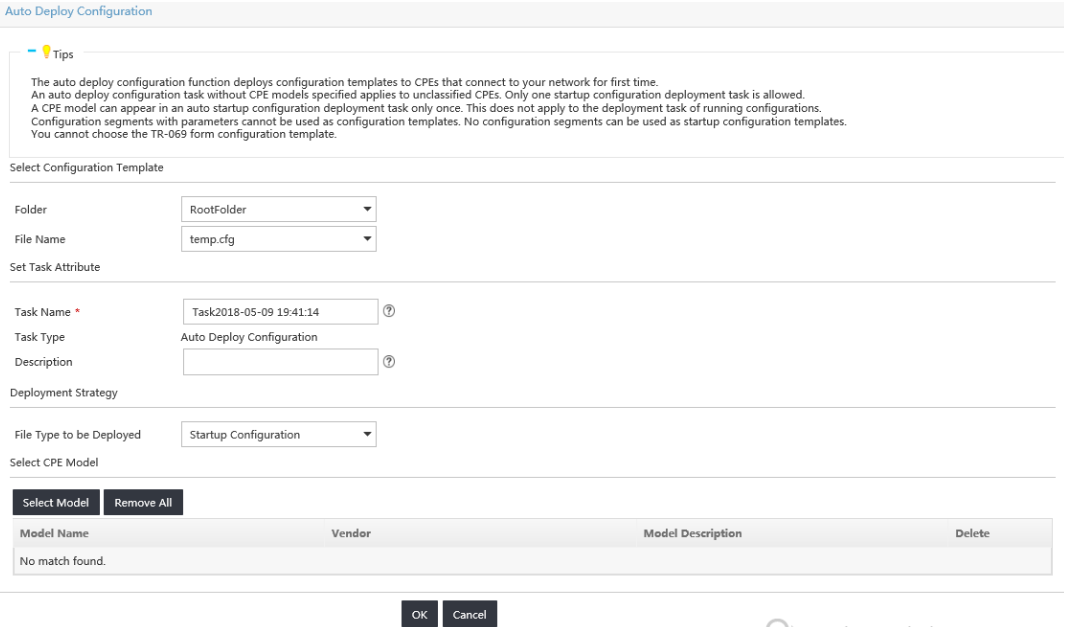
Service > BIMS > Configuration Management > Deployment Guide

b. Click **By CPE** from the **Auto Deployment Configuration** field.

c. Select a configuration template, select **Startup Configuration** from the **File Type to be Deployed** list, and click **Select Model** to select CPEs in Room A. Then, click **OK**.

You can search for CPEs by CPE group.

Figure 15 Auto deployment configuration



Auto Deploy Configuration

Tips

The auto deploy configuration function deploys configuration templates to CPEs that connect to your network for first time.
An auto deploy configuration task without CPE models specified applies to unclassified CPEs. Only one startup configuration deployment task is allowed.
A CPE model can appear in an auto startup configuration deployment task only once. This does not apply to the deployment task of running configurations.
Configuration segments with parameters cannot be used as configuration templates. No configuration segments can be used as startup configuration templates.
You cannot choose the TR-069 form configuration template.

Select Configuration Template

Folder: RootFolder

File Name: temp.cfg

Set Task Attribute

Task Name *: Task2018-05-09 19:41:14

Task Type: Auto Deploy Configuration

Description:

Deployment Strategy

File Type to be Deployed: Startup Configuration

Select CPE Model

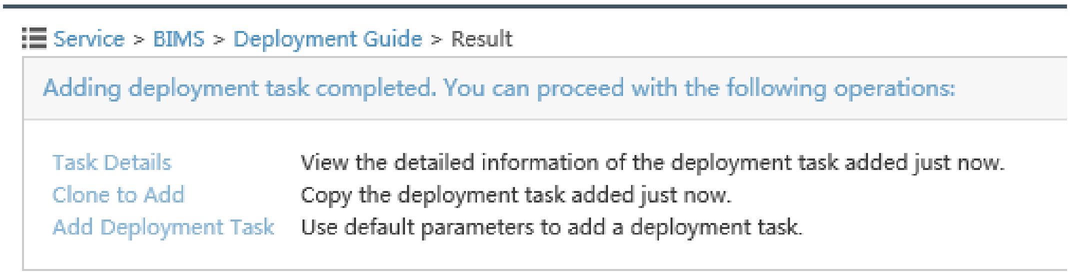
Select Model | Remove All

Model Name	Vendor	Model Description	Delete
No match found.			

OK | Cancel

d. Click **OK** on the **Auto Deploy Configuration** page.

Figure 16 Operation result



Service > BIMS > Deployment Guide > Result

Adding deployment task completed. You can proceed with the following operations:

Task Details	View the detailed information of the deployment task added just now.
Clone to Add	Copy the deployment task added just now.
Add Deployment Task	Use default parameters to add a deployment task.

e. Repeat the previous steps to add a deployment task for CPEs in Room B.

Configuring the DHCP server

1. Configure an IP address pool to assign IP addresses and DNS server address to the CPEs. This example uses subnet 10.185.10.0/24 for IP address assignment.

# Enable DHCP.

```
<DHCP_server> system-view
[DHCP_server] dhcp enable
```

# Enable DHCP server on VLAN-interface 1.

```
[DHCP_server] interface vlan-interface 1
[DHCP_server-Vlan-interface1] dhcp select server
[DHCP_server-Vlan-interface1] quit
```

# Exclude the DNS server address 10.185.10.60 and the ACS IP address 10.185.10.41 from dynamic allocation.

```
[DHCP_server] dhcp server forbidden-ip 10.185.10.41
[DHCP_server] dhcp server forbidden-ip 10.185.10.60
```

# Create DHCP address pool 0.

```
[DHCP_server] dhcp server ip-pool 0
```

# Assign subnet 10.185.10.0/24 to the address pool, and specify the DNS server address 10.185.10.60 in the address pool.

```
[DHCP_server-dhcp-pool-0] network 10.185.10.0 mask 255.255.255.0
[DHCP_server-dhcp-pool-0] dns-list 10.185.10.60
```

2. Configure DHCP Option 43 to contain the ACS URL, username, and password in hexadecimal format.

```
[DHCP_server-dhcp-pool-0] option 43 hex
013B687474703A2F2F6163732E64617461626173653A393039302F616373207669636B79203132333
435
```

Configuring the DNS server

Map http://acs.database:9090 to http://10.185.1.41:9090 on the DNS server. For more information about DNS configuration, see DNS server documentation.

Connecting the CPEs to the network

# Connect CPE 1 to the network, and then power on the CPE. (Details not shown.)

# Log in to CPE 1 and configure its interface GigabitEthernet 1/0/1 to use DHCP for IP address acquisition. At startup, the CPE obtains the IP address and ACS information from the DHCP server to initiate a connection to the ACS. After the connection is established, the CPE interacts with the ACS to complete autoconfiguration.

```
<CPE1> system-view
[CPE1] interface gigabitethernet 1/0/1
[CPE1-GigabitEthernet1/0/1] ip address dhcp-alloc
```

# Repeat the previous steps to configure the other CPEs.

Verifying the configuration

# Execute the **display current-configuration** command to verify that the running configurations on CPEs are the same as the configurations issued by the ACS.

Contents

Configuring SNMP

About SNMP

Simple Network Management Protocol (SNMP) is used for a management station to access and operate the devices on a network, regardless of their vendors, physical characteristics, and interconnect technologies.

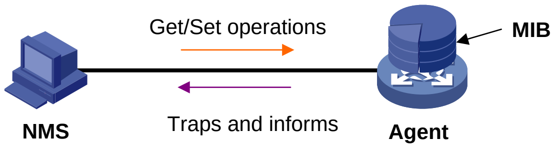
SNMP enables network administrators to read and set the variables on managed devices for state monitoring, troubleshooting, statistics collection, and other management purposes.

SNMP framework

The SNMP framework contains the following elements:

- **SNMP manager**—Works on an NMS to monitor and manage the SNMP-capable devices in the network. It can get and set values of MIB objects on the agent.
- **SNMP agent**—Works on a managed device to receive and handle requests from the NMS, and sends notifications to the NMS when events, such as an interface state change, occur.
- **Management Information Base (MIB)**—Specifies the variables (for example, interface status and CPU usage) maintained by the SNMP agent for the SNMP manager to read and set.

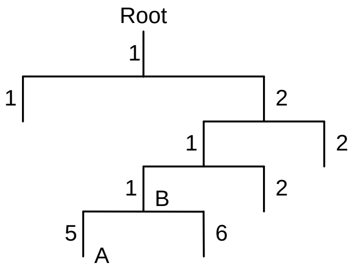
Figure 1 Relationship between NMS, agent, and MIB



MIB and view-based MIB access control

A MIB stores variables called "nodes" or "objects" in a tree hierarchy and identifies each node with a unique OID. An OID is a dotted numeric string that uniquely identifies the path from the root node to a leaf node. For example, object B in Figure 2 is uniquely identified by the OID {1.2.1.1}.

Figure 2 MIB tree



A MIB view represents a set of MIB objects (or MIB object hierarchies) with certain access privileges and is identified by a view name. The MIB objects included in the MIB view are accessible while those excluded from the MIB view are inaccessible.

A MIB view can have multiple view records each identified by a *view-name oid-tree* pair.

You control access to the MIB by assigning MIB views to SNMP groups or communities.

SNMP operations

SNMP provides the following basic operations:

- **Get**—NMS retrieves the value of an object node in an agent MIB.
- **Set**—NMS modifies the value of an object node in an agent MIB.
- **Notification**—SNMP notifications include traps and informs. The SNMP agent sends traps or informs to report events to the NMS. The difference between these two types of notification is that informs require acknowledgment but traps do not. Informs are more reliable but are also resource-consuming. Traps are available in SNMPv1, SNMPv2c, and SNMPv3. Informs are available only in SNMPv2c and SNMPv3.

Protocol versions

The device supports SNMPv1, SNMPv2c, and SNMPv3. An NMS and an SNMP agent must use the same SNMP version to communicate with each other.

- **SNMPv1**—Uses community names for authentication. To access an SNMP agent, an NMS must use the same community name as set on the SNMP agent. If the community name used by the NMS differs from the community name set on the agent, the NMS cannot establish an SNMP session to access the agent or receive traps from the agent.
- **SNMPv2c**—Uses community names for authentication. SNMPv2c is compatible with SNMPv1, but supports more operation types, data types, and error codes.
- **SNMPv3**—Uses a user-based security model (USM) to secure SNMP communication. You can configure authentication and privacy mechanisms to authenticate and encrypt SNMP packets for integrity, authenticity, and confidentiality.

Access control modes

SNMP uses the following modes to control access to MIB objects:

- **View-based Access Control Model**—VACM mode controls access to MIB objects by assigning MIB views to SNMP communities or users.
- **Role based access control**—RBAC mode controls access to MIB objects by assigning user roles to SNMP communities or users.
  - SNMP communities or users with predefined user role network-admin, context-admin, or level-15 have read and write access to all MIB objects.
  - SNMP communities or users with predefined user role network-operator or context-operator have read-only access to all MIB objects.
  - SNMP communities or users with a user-defined user role have access rights to MIB objects as specified by the **rule** command.

RBAC mode controls access on a per MIB object basis, and VACM mode controls access on a MIB view basis. As a best practice to enhance MIB security, use the RBAC mode.

If you create the same SNMP community or user with both modes multiple times, the most recent configuration takes effect. For more information about RBAC, see *Fundamentals Command Reference*.

SNMP silence

SNMP silence enables the device to automatically detect and defend against SNMP attacks.

After you enable SNMP, the device automatically starts an SNMP silence timer and counts the number of packets that fail SNMP authentication within 1 minute.

- If the number of the packets is smaller than 100, the device restarts the timer and counting.
- If the number of the packets is equal to or greater than 100, the SNMP module enters a silence period of 4 to 5 minutes, during which the device does not respond to any SNMP packets. After the silence period expires, the device restarts the timer and counting.

SNMP tasks at a glance

To configure SNMP, perform the following tasks:

1. Enabling the SNMP agent
2. Enabling SNMP versions
3. Configuring SNMP basic parameters
   - (Optional.) Configuring SNMP common parameters
   - Configuring an SNMPv1 or SNMPv2c community
   - Configuring an SNMPv3 group and user
4. (Optional.) Configuring SNMP notifications
5. (Optional.) Configuring SNMP logging
6. (Optional.) Enabling VA interface query and configuration by using MIB objects

Enabling the SNMP agent

Restrictions and guidelines

The SNMP agent is enabled when you use any command that begins with **snmp-agent** except for the **snmp-agent calculate-password** command.

The SNMP agent will fail to be enabled when the port that the agent will listen on is used by another service. You can use the **snmp-agent port** command to specify a listening port. To view the UDP port use information, execute the **display udp verbose** command. For more information about the **display udp verbose** command, see IP performance optimization commands in *Layer 3—IP Services Configuration Guide*.

If you disable the SNMP agent, the SNMP settings do not take effect. The **display current-configuration** command does not display the SNMP settings and the SNMP settings will not be saved in the configuration file. For the SNMP settings to take effect, enable the SNMP agent.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the SNMP agent.

   ```
   snmp-agent
   ```

   By default, the SNMP agent is disabled.

Enabling SNMP versions

Restrictions and guidelines

The devices supports SNMPv1, SNMPv2c, and SNMPv3. An NMS and an SNMP agent must use the same SNMP version to communicate with each other.

The community name and data carried in SNMPv1 and SNMPv2c messages are in plaintext form, putting the SNMP communication at risks. As a best practice, use SNMPv3.

SNMP notifications over IPv6 is supported only when you specify SNMPv2c or SNMPv3.

Procedure

1. Enter system view.

   system-view

2. Specify the SNMP version.

   snmp-agent sys-info version { **all** | { **v1** | **v2c** | **v3** } * }

   By default, SNMPv3 is enabled.

   If you execute the command multiple times with different options, all the configurations take effect, but only one SNMP version is used by the agent and NMS for communication.

Configuring SNMP common parameters

Restrictions and guidelines

An SNMP engine ID uniquely identifies a device in an SNMP managed network. Make sure the local SNMP engine ID is unique within your SNMP managed network to avoid communication problems. By default, the device is assigned a unique SNMP engine ID.

If you have configured SNMPv3 users, change the local SNMP engine ID only when necessary. The change can void the SNMPv3 usernames and encrypted keys you have configured.

The SNMP agent will fail to be enabled when the port that the agent will listen on is used by another service. You can use the **snmp-agent port** command to change the SNMP listening port. As a best practice, execute the **display udp verbose** command to view the UDP port use information before specifying a new SNMP listening port. For more information about the **display udp verbose** command, see IP performance optimization commands in *Layer 3—IP Services Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Specify an SNMP listening port.

   snmp-agent port *port-number*

   By default, the SNMP listening port is UDP port 161.

3. Set a local SNMP engine ID.

   snmp-agent local-engineid *engineid*

   By default, the local SNMP engine ID is the company ID plus the device ID.

4. Set an engine ID for a remote SNMP entity.

   snmp-agent remote { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*] **engineid** *engineid*

   By default, no remote entity engine IDs exist.

   This step is required for the device to send SNMPv3 notifications to a host, typically NMS.

5. Create or update a MIB view.

   snmp-agent mib-view { **excluded** | **included** } *view-name oid-tree* [**mask** *mask-value*]

   By default, the MIB view **ViewDefault** is predefined. In this view, all the MIB objects in the **iso** subtree but the **snmpUsmMIB**, **snmpVacmMIB**, and **snmpModules.18** subtrees are accessible.

Each *view-name oid-tree* pair represents a view record. If you specify the same record with different MIB sub-tree masks multiple times, the most recent configuration takes effect.

6. Configure the system management information.

- Configure the system contact.

  snmp-agent sys-info contact *sys-contact*

  By default, the system contact is **NSFOCUS**

- Configure the system location.

  snmp-agent sys-info location *sys- location*

  By default, the system location is **Beijing, China**.

7. Create an SNMP context.

snmp-agent context *context-name*

By default, no SNMP contexts exist.

8. Configure the maximum SNMP packet size (in bytes) that the SNMP agent can handle.

snmp-agent packet max-size *byte-count*

By default, an SNMP agent can process SNMP packets with a maximum size of 1500 bytes.

Configuring an SNMPv1 or SNMPv2c community

About configuring an SNMPv1 or SNMPv2c community

You can create an SNMPv1 or SNMPv2c community by using a community name or by creating an SNMPv1 or SNMPv2c user. After you create an SNMPv1 or SNMPv2c user, the system automatically creates a community by using the username as the community name.

Restrictions and guidelines for configuring an SNMPv1 or SNMPv2c community

Make sure the NMS and agent use the same SNMP community name.

Only users with the network-admin, context-admin, or level-15 user role can create SNMPv1 or SNMPv2c communities, users, or groups. Users with other user roles cannot create SNMPv1 or SNMPv2c communities, users, or groups even if these roles are granted access to related commands or commands of the SNMPv1 or SNMPv2c feature.

Configuring an SNMPv1/v2c community by a community name

1. Enter system view.

system-view

2. Create an SNMPv1/v2c community. Choose one option as needed.

- In VACM mode:

  snmp-agent community { **read** | **write** } [**simple** | **cipher**] *community-name* [**mib-view** *view-name*] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } | **acl ipv6** { *ipv6-acl-number* | **name** *ipv6-acl-name* }] *

- In RBAC mode:

```
snmp-agent community [ simple | cipher ] community-name user-role
role-name [ acl { ipv4-acl-number | name ipv4-acl-name } | acl ipv6
{ ipv6-acl-number | name ipv6-acl-name } ] *
```

3. (Optional.) Map the SNMP community name to an SNMP context.

```
snmp-agent community-map community-name context context-name
```

Configuring an SNMPv1/v2c community by creating an SNMPv1/v2c user

1. Enter system view.

```
system-view
```

2. Create an SNMPv1/v2c group.

```
snmp-agent group { v1 | v2c } group-name [ notify-view view-name |
read-view view-name | write-view view-name ] * [ acl { ipv4-acl-number |
name ipv4-acl-name } | acl ipv6 { ipv6-acl-number | name
ipv6-acl-name } ] *
```

3. Add an SNMPv1/v2c user to the group.

```
snmp-agent usm-user { v1 | v2c } user-name group-name [ acl
{ ipv4-acl-number | name ipv4-acl-name } | acl ipv6 { ipv6-acl-number
| name ipv6-acl-name } ] *
```

The system automatically creates an SNMP community by using the username as the community name.

4. (Optional.) Map the SNMP community name to an SNMP context.

```
snmp-agent community-map community-name context context-name
```

Configuring an SNMPv3 group and user

Restrictions and guidelines for configuring an SNMPv3 group and user

Only users with the network-admin, context-admin, or level-15 user role can create SNMPv3 users or groups.Users with other user roles cannot create SNMPv3 users or groups even if these roles are granted access to related commands or commands of the SNMPv3 feature.

SNMPv3 users are managed in groups. All SNMPv3 users in a group share the same security model, but can use different authentication and encryption algorithms and keys. Table 1 describes the basic configuration requirements for different security models.

Table 1 Basic configuration requirements for different security models

Security model	Keyword for the group	Parameters for the user	Remarks
Authentication with privacy	`privacy`	Authentication and encryption algorithms and keys	For an NMS to access the agent, make sure the NMS and agent use the same authentication and encryption keys.

Security model	Keyword for the group	Parameters for the user	Remarks
Authentication without privacy	**authentication**	Authentication algorithm and key	For an NMS to access the agent, make sure the NMS and agent use the same authentication key.
No authentication, no privacy	N/A	N/A	The authentication and encryption keys, if configured, do not take effect.

Configuring an SNMPv3 group and user

1. Enter system view.

   system-view

2. Create an SNMPv3 group.

   snmp-agent group v3 *group-name* [**authentication** | **privacy**] [**notify-view** *view-name* | **read-view** *view-name* | **write-view** *view-name*] * [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } | **acl ipv6** { *ipv6-acl-number* | **name** *ipv6-acl-name* }] *

3. (Optional.) Calculate the encrypted form for the key in plaintext form.

   snmp-agent calculate-password *plain-password* **mode** { **3desmd5** | **3dessha** | **md5** | **sha** } { **local-engineid** | **specified-engineid** *engineid* }

4. Create an SNMPv3 user. Choose one option as needed.

   - In VACM mode:

     snmp-agent usm-user v3 *user-name group-name* [**remote** { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]] [{ **cipher** | **simple** } **authentication-mode** { **md5** | **sha** } *auth-password* [**privacy-mode** { **3des** | **aes128** | **des56** } *priv-password*]] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } | **acl ipv6** { *ipv6-acl-number* | **name** *ipv6-acl-name* }] *

   - In RBAC mode:

     snmp-agent usm-user v3 *user-name* **user-role** *role-name* [**remote** { *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]] [{ **cipher** | **simple** } **authentication-mode** { **md5** | **sha** } *auth-password* [**privacy-mode** { **3des** | **aes128** | **des56** } *priv-password*]] [**acl** { *ipv4-acl-number* | **name** *ipv4-acl-name* } | **acl ipv6** { *ipv6-acl-number* | **name** *ipv6-acl-name* }] *

   To send notifications to an SNMPv3 NMS, you must specify the **remote** keyword.

5. (Optional.) Assign a user role to the SNMPv3 user created in RBAC mode.

   snmp-agent usm-user v3 *user-name* **user-role** *role-name*

   By default, an SNMPv3 user has the user role assigned to it at its creation.

Configuring SNMP notifications

About SNMP notifications

The SNMP agent sends notifications (traps and informs) to inform the NMS of significant events, such as link state changes and user logins or logouts. After you enable notifications for a module, the module sends the generated notifications to the SNMP agent. The SNMP agent sends the received notifications as traps or informs based on the current configuration. Unless otherwise stated, the **trap** keyword in the command line includes both traps and informs.

Enabling SNMP notifications

Restrictions and guidelines

Enable SNMP notifications only if necessary. SNMP notifications are memory-intensive and might affect device performance.

To generate linkUp or linkDown notifications when the link state of an interface changes, you must perform the following tasks:

- Enable linkUp or linkDown notification globally by using the **snmp-agent trap enable standard** [**linkdown** | **linkup**] * command.
- Enable linkUp or linkDown notification on the interface by using the **enable snmp trap updown** command.

After you enable notifications for a module, whether the module generates notifications also depends on the configuration of the module. For more information, see the configuration guide for each module.

SNMP notifications over IPv6 is supported only when you specify SNMPv2c or SNMPv3.

Procedure

1. Enter system view.

   system-view
2. Enable SNMP notifications.

   snmp-agent trap enable [**configuration** | *protocol* | **standard** [**authentication** | **coldstart** | **linkdown** | **linkup** | **warmstart**] * | **system**]

   By default, SNMP configuration notifications, standard notifications, and system notifications are enabled. Whether other SNMP notifications are enabled varies by modules.

   To enable the device to send SNMP notifications for a protocol, first enable the protocol.
3. Enter interface view.

   interface *interface-type interface-number*
4. Enable link state notifications.

   enable snmp trap updown

   By default, link state notifications are enabled.

Configuring parameters for sending SNMP notifications

About this task

You can configure the SNMP agent to send notifications as traps or informs to a host, typically an NMS, for analysis and management. Traps are less reliable and use fewer resources than informs, because an NMS does not send an acknowledgment when it receives a trap.

When network congestion occurs or the destination is not reachable, the SNMP agent buffers notifications in a queue. You can set the queue size and the notification lifetime (the maximum time that a notification can stay in the queue). A notification is deleted when its lifetime expires. When the notification queue is full, the oldest notifications are automatically deleted.

By default, the device monitors the values of the variables in a trap. When the values reach the specified thresholds, the device enters alarm state and sends a trap to the NMS, The device sends a trap only when it enters alarm state or alarm removed state. The trap retransmission mechanism enables the device to send traps at the specified intervals to notify the NMS as long as the device is in alarm state. Only CPU usage and memory usage traps are supported.

Configuring the parameters for sending SNMP traps

1. Enter system view.

   system-view

2. Configure a target host.

   snmp-agent target-host trap address udp-domain { *ipv4-target-host* | **ipv6** *ipv6-target-host* } [**source-ip** *source-ip-address* | **udp-port** *port-number* | **vpn-instance** *vpn-instance-name*] * **params securityname** *security-string* [**v1** | **v2c** | **v3** [**authentication** | **privacy**]]

   By default, no target host is configured.

3. (Optional.) Configure a source address for sending traps.

   snmp-agent trap source *interface-type* { *interface-number* | *interface-number.subnumber* }

   By default, SNMP uses the IP address of the outgoing routed interface as the source IP address.

   For a trap sent to a particular NMS, the source IP address specified by using the **snmp-agent target-host** command overrides that specified by using this command.

Configuring the parameters for sending SNMP informs

1. Enter system view.

   system-view

2. Configure a target host.

   snmp-agent target-host inform address udp-domain { *ipv4-target-host* | **ipv6** *ipv6-target-host* } [**source-ip** *source-ip-address* | **udp-port** *port-number* | **vpn-instance** *vpn-instance-name*] ***params securityname** *security-string* { **v2c** | **v3** [**authentication** | **privacy**] }

   By default, no target host is configured.

   Only SNMPv2c and SNMPv3 support inform packets.

3. (Optional.) Configure a source address for sending informs.

   snmp-agent inform source *interface-type* { *interface-number* | *interface-number.subnumber* }

   By default, SNMP uses the IP address of the outgoing routed interface as the source IP address.

For an inform sent to a particular NMS, the source IP address specified by using the **snmp-agent target-host** command overrides that specified by using this command.

Configuring common parameters for sending notifications

1. Enter system view.

   system-view

2. (Optional.) Enable extended linkUp/linkDown notifications.

   snmp-agent trap if-mib link extended

   By default, the SNMP agent sends standard linkUp/linkDown notifications.

   An extended linkUp/linkDown notification adds the interface name, interface type, and interface description to the linkUp/linkDown notification. If the NMS does not support extended linkUp/linkDown notifications, do not use this command.

3. (Optional.) Set the notification queue size.

   snmp-agent trap queue-size *size*

   By default, the notification queue can hold 100 notification messages.

4. (Optional.) Set the notification lifetime.

   snmp-agent trap life *seconds*

   The default notification lifetime is 120 seconds.

Configuring SNMP logging

About this task

The SNMP agent logs Get requests, Set requests, Set responses, SNMP notifications, and SNMP authentication failures, but does not log Get responses.

- **Get operation**—The agent logs the IP address of the NMS, name of the accessed node, and node OID.
- **Set operation**—The agent logs the NMS' IP address, name of accessed node, node OID, variable value, and error code and index for the Set operation.
- **Notification tracking**—The agent logs the SNMP notifications after sending them to the NMS.
- **SNMP authentication failure**—The agent logs related information when an NMS fails to be authenticated by the agent.

The SNMP module sends these logs to the information center. You can configure the information center to output these messages to certain destinations, such as the console and the log buffer. The total output size for the node field (MIB node name) and the value field (value of the MIB node) in each log entry is 1024 bytes. If this limit is exceeded, the information center truncates the data in the fields. For more information about the information center, see "Configuring the information center."

Restrictions and guidelines

Enable SNMP logging only if necessary. SNMP logging is memory-intensive and might impact device performance.

Procedure

1. Enter system view.

   system-view

2. Enable SNMP logging.

   snmp-agent log { **all** | **authfail** | **get-operation** | **set-operation** }

   By default, SNMP logging is disabled.

3. Enable SNMP notification logging.

snmp-agent trap log

By default, SNMP notification logging is disabled.

Enabling VA interface query and configuration by using MIB objects

About this task

By default, VA interfaces cannot be queried and configured by using MIB objects. The device ignores the query and configuration requests from the NMS for VA interfaces. This not only enhances the device efficiency to obtain other interface information and improves user experiences, but also reduces the device workload and avoids waster of CPU resources.

To enable VA interface query and configuration by using MIB objects, execute this command.

For more information about VA interfaces, see PPP configuration in *Layer 2—WAN Access Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable VA interface query and configuration by using MIB objects.

   snmp virtual-access visible

   By default, VA interfaces cannot be queried and configured by using MIB objects.

Display and maintenance commands for SNMP

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display SNMPv1 or SNMPv2c community information.	**display snmp-agent community** [**read** | **write**]
Display SNMP contexts.	**display snmp-agent context** [*context-name*]
Display SNMP group information.	**display snmp-agent group** [*group-name*]
Display the local engine ID.	**display snmp-agent local-engineid**
Display SNMP MIB node information.	**display snmp-agent mib-node** [**details** | **index-node** | **trap-node** | **verbose**]
Display MIB view information.	**display snmp-agent mib-view** [**exclude** | **include** | **viewname** *view-name*]
Display remote engine IDs.	**display snmp-agent remote** [{ *ipv4-address* | **ipv6** *ipv6-address* } [**vpn-instance** *vpn-instance-name*]]
Display SNMP agent statistics.	**display snmp-agent statistics**

Task	Command
Display SNMP agent system information.	**display snmp-agent sys-info** [**contact** | **location** | **version**] *
Display basic information about the notification queue.	**display snmp-agent trap queue**
Display SNMP notifications drop records.	**display snmp-agent trapbuffer drop**
Display SNMP notifications sending records.	**display snmp-agent trapbuffer send**
Display SNMP notifications enabling status for modules.	**display snmp-agent trap-list**
Display SNMPv3 user information.	**display snmp-agent usm-user** [**engineid** *engineid* | **username** *user-name* | **group** *group-name*] *
Clear all records from the SNMP trap buffer.	**reset snmp-agent trapbuffer**

Contents

Configuring RMON

About RMON

Remote Network Monitoring (RMON) is an SNMP-based network management protocol. It enables proactive remote monitoring and management of network devices.

RMON working mechanism

RMON can periodically or continuously collect traffic statistics for an Ethernet port and monitor the values of MIB objects on a device. When a value reaches the threshold, the device automatically logs the event or sends a notification to the NMS. The NMS does not need to constantly poll MIB variables and compare the results.

RMON uses SNMP notifications to notify NMSs of various alarm conditions. SNMP reports function and interface operating status changes such as link up, link down, and module failure to the NMS.

RMON groups

Among standard RMON groups, the device implements the statistics group, history group, event group, alarm group, probe configuration group, and user history group. The NF system also implements a private alarm group, which enhances the standard alarm group. The probe configuration group and user history group are not configurable from the CLI. To configure these two groups, you must access the MIB.

Statistics group

The statistics group samples traffic statistics for monitored Ethernet interfaces and stores the statistics in the Ethernet statistics table (ethernetStatsTable). The statistics include:

- Number of collisions.
- CRC alignment errors.
- Number of undersize or oversize packets.
- Number of broadcasts.
- Number of multicasts.
- Number of bytes received.
- Number of packets received.

The statistics in the Ethernet statistics table are cumulative sums.

History group

The history group periodically samples traffic statistics on interfaces and saves the history samples in the history table (etherHistoryTable). The statistics include:

- Bandwidth utilization.
- Number of error packets.
- Total number of packets.

The history table stores traffic statistics collected for each sampling interval.

Event group

The event group controls the generation and notifications of events triggered by the alarms defined in the alarm group and the private alarm group. The following are RMON alarm event handling methods:

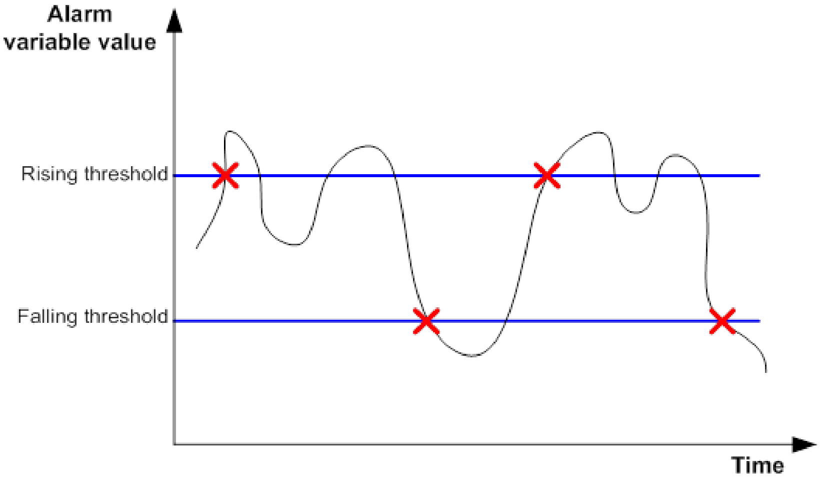
- **Log**—Logs event information (including event time and description) in the event log table so the management device can get the logs through SNMP.
- **Trap**—Sends an SNMP notification when the event occurs.
- **Log-Trap**—Logs event information in the event log table and sends an SNMP notification when the event occurs.
- **None**—Takes no actions.

Alarm group

The RMON alarm group monitors alarm variables, such as the count of incoming packets (etherStatsPkts) on an interface. After you create an alarm entry, the RMON agent samples the value of the monitored alarm variable regularly. If the value of the monitored variable is greater than or equal to the rising threshold, a rising alarm event is triggered. If the value of the monitored variable is smaller than or equal to the falling threshold, a falling alarm event is triggered. The event group defines the action to take on the alarm event.

If an alarm entry crosses a threshold multiple times in succession, the RMON agent generates an alarm event only for the first crossing. For example, if the value of a sampled alarm variable crosses the rising threshold multiple times before it crosses the falling threshold, only the first crossing triggers a rising alarm event, as shown in Figure 1.

Figure 1 Rising and falling alarm events



Private alarm group

The private alarm group enables you to perform basic math operations on multiple variables, and compare the calculation result with the rising and falling thresholds.

The RMON agent samples variables and takes an alarm action based on a private alarm entry as follows:

1. Samples the private alarm variables in the user-defined formula.
2. Processes the sampled values with the formula.
3. Compares the calculation result with the predefined thresholds, and then takes one of the following actions:
    - Triggers the event associated with the rising alarm event if the result is equal to or greater than the rising threshold.
    - Triggers the event associated with the falling alarm event if the result is equal to or less than the falling threshold.

If a private alarm entry crosses a threshold multiple times in succession, the RMON agent generates an alarm event only for the first crossing. For example, if the value of a sampled alarm variable crosses the rising threshold multiple times before it crosses the falling threshold, only the first crossing triggers a rising alarm event.

Sample types for the alarm group and the private alarm group

The RMON agent supports the following sample types:

- **absolute**—RMON compares the value of the monitored variable with the rising and falling thresholds at the end of the sampling interval.
- **delta**—RMON subtracts the value of the monitored variable at the previous sample from the current value, and then compares the difference with the rising and falling thresholds.

Protocols and standards

- RFC 4502, *Remote Network Monitoring Management Information Base Version 2*
- RFC 2819, *Remote Network Monitoring Management Information Base Status of this Memo*

Configuring the RMON statistics function

About the RMON statistics function

RMON implements the statistics function through the Ethernet statistics group and the history group.

The Ethernet statistics group provides the cumulative statistic for a variable from the time the statistics entry is created to the current time.

The history group provides statistics that are sampled for a variable for each sampling interval. The history group uses the history control table to control sampling, and it stores samples in the history table.

Creating an RMON Ethernet statistics entry

Restrictions and guidelines

The index of an RMON statistics entry must be globally unique. If the index has been used by another interface, the creation operation fails.

You can create only one RMON statistics entry for an Ethernet interface.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Create an RMON Ethernet statistics entry.

   `rmon statistics` *`entry-number`* [**`owner`** *`text`*]

   By default, no RMON Ethernet statistics entry exists.

Creating an RMON history control entry

Restrictions and guidelines

You can configure multiple history control entries for one interface, but you must make sure their entry numbers and sampling intervals are different.

You can create a history control entry successfully even if the specified bucket size exceeds the available history table size. RMON will set the bucket size as closely to the expected bucket size as possible.

Procedure

1. Enter system view.

   `system-view`

2. Enter Ethernet interface view.

   `interface` *`interface-type interface-number`*

3. Create an RMON history control entry.

   `rmon history` *`entry-number`* **`buckets`** *`number`* **`interval`** *`interval`* [**`owner`** *`text`*]

   By default, no RMON history control entries exist.

   You can create multiple RMON history control entries for an Ethernet interface.

Configuring the RMON alarm function

Restrictions and guidelines

When you create a new event, alarm, or private alarm entry, follow these restrictions and guidelines:

- The entry must not have the same set of parameters as an existing entry.
- The maximum number of entries is not reached.

Table 1 shows the parameters to be compared for duplication and the entry limits.

Table 1 RMON configuration restrictions

Entry	Parameters to be compared	Maximum number of entries
Event	• Event description (**`description`** *`string`*) • Event type (**`log`**, **`trap`**, **`logtrap`**, or **`none`**) • Community name (*`security-string`*)	60
Alarm	• Alarm variable (*`alarm-variable`*) • Sampling interval (*`sampling-interval`*) • Sample type (**`absolute`** or **`delta`**) • Rising threshold (*`threshold-value1`*) • Falling threshold (*`threshold-value2`*)	60

Entry	Parameters to be compared	Maximum number of entries
Private alarm	• Alarm variable formula (*prialarm-formula*) • Sampling interval (*sampling-interval*) • Sample type (**absolute** or **delta**) • Rising threshold (*threshold-value1*) • Falling threshold (*threshold-value2*)	50

Prerequisites

To send notifications to the NMS when an alarm is triggered, configure the SNMP agent as described in "Configuring SNMP" before configuring the RMON alarm function.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Create an RMON event entry.

   rmon event *entry-number* [**description** *string*] { **log** | **log-trap** *security-string* | **none** | **trap** *security-string* } [**owner** *text*]

   By default, no RMON event entries exist.

3. Create an RMON alarm entry.

   - Create an RMON alarm entry.

     rmon alarm *entry-number alarm-variable sampling-interval* { **absolute** | **delta** } [**startup-alarm** { **falling** | **rising** | **rising-falling** }] **rising-threshold** *threshold-value1 event-entry1* **falling-threshold** *threshold-value2 event-entry2* [**owner** *text*]

   - Create an RMON private alarm entry.

     rmon prialarm *entry-number prialarm-formula prialarm-des sampling-interval* { **absolute** | **delta** } [**startup-alarm** { **falling** | **rising** | **rising-falling** }] **rising-threshold** *threshold-value1 event-entry1* **falling-threshold** *threshold-value2 event-entry2* **entrytype** { **forever** | **cycle** *cycle-period* } [**owner** *text*]

   By default, no RMON alarm entries or RMON private alarm entries exist.

   You can associate an alarm with an event that has not been created yet. The alarm will trigger the event only after the event is created.

Display and maintenance commands for RMON

Execute **display** commands in any view.

Task	Command
Display RMON alarm entries.	**display rmon alarm** [*entry-number*]
Display RMON event entries.	**display rmon event** [*entry-number*]
Display log information for event entries.	**display rmon eventlog** [*entry-number*]

Task	Command
Display RMON history control entries and history samples.	**display rmon history** [*interface-type interface-number*]
Display RMON private alarm entries.	**display rmon prialarm** [*entry-number*]
Display RMON statistics.	**display rmon statistics** [*interface-type interface-number*]

Contents

Configuring the Event MIB

About the Event MIB

The Event Management Information Base (Event MIB) is an SNMPv3-based network management protocol and is an enhancement to remote network monitoring (RMON). The Event MIB uses Boolean tests, existence tests, and threshold tests to monitor MIB objects on a local or remote system. It triggers the predefined notification or set action when a monitored object meets the trigger condition.

Trigger

The Event MIB uses triggers to manage and associate the three elements of the Event MIB: monitored object, trigger condition, and action.

Monitored objects

The Event MIB can monitor the following MIB objects:

- Table node.
- Conceptual row node.
- Table column node.
- Simple leaf node.
- Parent node of a leaf node.

To monitor a single MIB object, specify it by its OID or name. To monitor a set of MIB objects, specify the common OID or name of the group and enable wildcard matching. For example, specify ifDescr.2 to monitor the description for the interface with index 2. Specify ifDescr and enable wildcard matching to monitor the descriptions for all interfaces.

Trigger test

A trigger supports Boolean, existence, and threshold tests.

Boolean test

A Boolean test compares the value of the monitored object with the reference value and takes actions according to the comparison result. The comparison types include **unequal**, **equal**, **less**, **lessorequal**, **greater**, and **greaterorequal**. For example, if the comparison type is **equal**, an event is triggered when the value of the monitored object equals the reference value. The event will not be triggered again until the value becomes unequal and comes back to equal.

Existence test

An existence test monitors and manages the absence, presence, and change of a MIB object, for example, interface status. When a monitored object is specified, the system reads the value of the monitored object regularly.

- If the test type is **Absent**, the system triggers an alarm event and takes the specified action when the state of the monitored object changes to absent.
- If the test type is **Present**, the system triggers an alarm event and takes the specified action when the state of the monitored object changes to present.
- If the test type is **Changed**, the system triggers an alarm event and takes the specified action when the value of the monitored object changes.

Threshold test

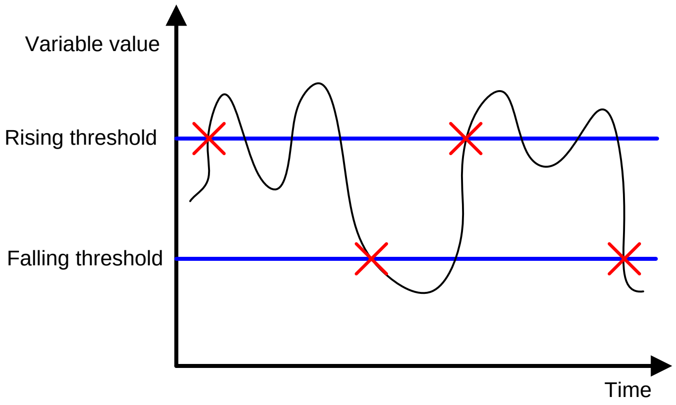
A threshold test regularly compares the value of the monitored object with the threshold values.

- A rising alarm event is triggered if the value of the monitored object is greater than or equal to the rising threshold.
- A falling alarm event is triggered if the value of the monitored object is smaller than or equal to the falling threshold.
- A rising alarm event is triggered if the difference between the current sampled value and the previous sampled value is greater than or equal to the delta rising threshold.
- A falling alarm event is triggered if the difference between the current sampled value and the previous sampled value is smaller than or equal to the delta falling threshold.
- A falling alarm event is triggered if the values of the monitored object, the rising threshold, and the falling threshold are the same.
- A falling alarm event is triggered if the delta rising threshold, the delta falling threshold, and the difference between the current sampled value and the previous sampled value is the same.

The alarm management module defines the set or notification action to take on alarm events.

If the value of the monitored object crosses a threshold multiple times in succession, the managed device triggers an alarm event only for the first crossing. For example, if the value of a sampled object crosses the rising threshold multiple times before it crosses the falling threshold, only the first crossing triggers a rising alarm event, as shown in Figure 1.

Figure 1 Rising and falling alarm events



Event actions

The Event MIB triggers one or both of the following actions when the trigger condition is met:

- **Set action**—Uses SNMP to set the value of the monitored object.
- **Notification action**—Uses SNMP to send a notification to the NMS. If an object list is specified for the notification action, the notification will carry the specified objects in the object list.

Object list

An object list is a set of MIB objects. You can specify an object list in trigger view, trigger-test view (including trigger-Boolean view, trigger existence view, and trigger threshold view), and action-notification view. If a notification action is triggered, the device sends a notification carrying the object list to the NMS.

If you specify an object list respectively in any two of the views or all the three views, the object lists are added to the triggered notifications in this sequence: trigger view, trigger-test view, and action-notification view.

Object owner

Trigger, event, and object list use an owner and name for unique identification. The owner must be an SNMPv3 user that has been created on the device. If you specify a notification action for a trigger, you must establish an SNMPv3 connection between the device and NMS by using the SNMPv3 username. For more information about SNMPv3 user, see "SNMP configuration".

Restrictions and guidelines: Event MIB configuration

The Event MIB and RMON are independent of each other. You can configure one or both of the features for network management.

You must specify the same owner for a trigger, object lists of the trigger, and events of the trigger.

Event MIB tasks at a glance

To configure the Event MIB, perform the following tasks:

1. Configuring the Event MIB global sampling parameters
2. (Optional.) Configuring Event MIB object lists

   Perform this task so that the device sends a notification that carries the specified object list to the NMS when a notification action is triggered.
3. Configuring an event

   The device supports set and notification actions. Choose one or two of the following actions:
   - Creating an event
   - Configuring a set action for an event
   - Configuring a notification action for an event
   - Enabling the event
4. Configuring a trigger

   A trigger supports Boolean, existence, and threshold tests. Choose one or more of the following tests:
   - Creating a trigger and configuring its basic parameters
   - Configuring a Boolean trigger test
   - Configuring an existence trigger test
   - Configuring a threshold trigger test
   - Enabling trigger sampling
5. (Optional.) Enabling SNMP notifications for the Event MIB module

Prerequisites for configuring the Event MIB

Before you configure the Event MIB, perform the following tasks:

- Create an SNMPv3 user. Assign the user the rights to read and set the values of the specified MIB objects and object lists.

- Make sure the SNMP agent and NMS are configured correctly and the SNMP agent can send notifications to the NMS correctly.

Configuring the Event MIB global sampling parameters

Restrictions and guidelines

This tasks takes effect only on monitored instances to be created.

Procedure

1. Enter system view.

   `system-view`

2. Set the minimum sampling interval.

   `snmp mib event sample minimum` *min-number*

   By default, the minimum sampling interval is 1 second.

   The sampling interval of a trigger must be greater than the minimum sampling interval.

3. Configure the maximum number of object instances that can be concurrently sampled.

   `snmp mib event sample instance maximum` *max-number*

   By default, the value is 0. The maximum number of object instances that can be concurrently sampled is limited by the available resources.

Configuring Event MIB object lists

About this task

Perform this task so that the device sends a notification that carries the specified objects to the NMS when a notification action is triggered.

Procedure

1. Enter system view.

   `system-view`

2. Configure an Event MIB object list.

   `snmp mib event object list owner` *group-owner* **`name`** *group-name object-index* **`oid`** *object-identifier* [**`wildcard`**]

   The object can be a table node, conceptual row node, table column node, simple leaf node, or parent node of a leaf node.

Configuring an event

Creating an event

1. Enter system view.

   `system-view`

2. Create an event and enter its view.

   `snmp mib event owner` *event-owner* **`name`** *event-name*

3. (Optional.) Configure a description for the event.

`description` *text*

By default, an event does not have a description.

Configuring a set action for an event

1. Enter system view.

   `system-view`

2. Enter event view.

   `snmp mib event owner` *event-owner* **name** *event-name*

3. Enable the set action and enter set action view.

   `action set`

   By default, no action is specified for an event.

4. Specify an object by its OID for the set action.

   `oid` *object-identifier*

   By default, no object is specified for a set action.

   The object can be a table node, conceptual row node, table column node, simple leaf node, or parent node of a leaf node.

5. Enable OID wildcarding.

   `wildcard oid`

   By default, OID wildcarding is disabled.

6. Set the value for the object.

   `value` *integer-value*

   The default value for the object is 0.

7. (Optional.) Specify a context for the object.

   `context` *context-name*

   By default, no context is specified for an object.

8. (Optional.) Enable context wildcarding.

   `wildcard context`

   By default, context wildcarding is disabled.

   A wildcard context contains the specified context and the wildcarded part.

Configuring a notification action for an event

1. Enter system view.

   `system-view`

2. Enter event view

   `snmp mib event owner` *event-owner* **name** *event-name*

3. Enable the notification action and enter notification action view.

   `action notification`

   By default, no action is specified for an event.

4. Specify an object to execute the notification action by its OID.

   `oid` *object-identifier*

   By default, no object is specified for executing the notification action.

   The object must be a notification object.

5. Specify an object list to be added to the notification triggered by the event.

   `object list owner` *group-owner* **`name`** *group-name*

   By default, no object list is specified for the notification action.

   If you do not specify an object list for the notification action or the specified object list does not contain variables, no variables will be carried in the notification.

Enabling the event

Restrictions and guidelines

The Boolean, existence, and threshold events can be triggered only after you perform this task.

To change an enabled event, first disable the event.

Procedure

1. Enter system view.

   `system-view`

2. Enter event view.

   `snmp mib event owner` *`event-owner`* **`name`** *`event-name`*

3. Enable the event.

   `event enable`

   By default, an event is disabled.

Configuring a trigger

Creating a trigger and configuring its basic parameters

1. Enter system view.

   `system-view`

2. Create a trigger and enter its view.

   `snmp mib event trigger owner` *trigger-owner* **`name`** *trigger-name*

   The trigger owner must be an existing SNMPv3 user.

3. (Optional.) Configure a description for the trigger.

   `description` *text*

   By default, a trigger does not have a description.

4. Set a sampling interval for the trigger.

   `frequency` *interval*

   By default, the sampling interval is 600 seconds.

   Make sure the sampling interval is greater than or equal to the Event MIB minimum sampling interval.

5. Specify a sampling method.

   `sample` { **`absolute`** | **`delta`** }

   The default sampling method is **`absolute`**.

6. Specify an object to be sampled by its OID.

   `oid` *object-identifier*

   By default, the OID is 0.0. No object is specified for a trigger.

   If you execute this command multiple times, the most recent configuration takes effect.

7. (Optional.) Enable OID wildcarding.

   `wildcard oid`

   By default, OID wildcarding is disabled.

8. (Optional.) Configure a context for the monitored object.

   `context` *context-name*

   By default, no context is configured for a monitored object.

9. (Optional.) Enable context wildcarding.

   `wildcard context`

   By default, context wildcarding is disabled.

10. (Optional.) Specify the object list to be added to the triggered notification.

    `object list owner` *group-owner* **name** *group-name*

    By default, no object list is specified for a trigger.

Configuring a Boolean trigger test

1. Enter system view.

   `system-view`

2. Enter trigger view.

   `snmp mib event trigger owner` *trigger-owner* **name** *trigger-name*

3. Specify a Boolean test for the trigger and enter trigger-Boolean view.

   `test boolean`

   By default, no test is configured for a trigger.

4. Specify a Boolean test comparison type.

   `comparison` { **`equal`** | **`greater`** | **`greaterorequal`** | **`less`** | **`lessorequal`** | **`unequal`** }

   The default Boolean test comparison type is **`unequal`**.

5. Set a reference value for the Boolean trigger test.

   `value` *integer-value*

   The default reference value for a Boolean trigger test is 0.

6. Specify an event for the Boolean trigger test.

   `event owner` *event-owner* **name** *event-name*

   By default, no event is specified for a Boolean trigger test.

7. (Optional.) Specify the object list to be added to the notification triggered by the test.

   `object list owner` *group-owner* **name** *group-name*

   By default, no object list is specified for a Boolean trigger test.

8. Enable the event to be triggered when the trigger condition is met at the first sampling.

   `startup enable`

   By default, the event is triggered when the trigger condition is met at the first sampling.

   Before the first sampling, you must enable this command to allow the event to be triggered.

Configuring an existence trigger test

1. Enter system view.

   `system-view`

2. Enter trigger view.

   snmp mib event trigger owner *trigger-owner* **name** *trigger-name*

3. Specify an existence test for the trigger and enter trigger-existence view.

   test existence

   By default ,no test is configured for a trigger.

4. Specify an event for the existence trigger test.

   event owner *event-owner* **name** *event-name*

   By default, no event is specified for an existence trigger test.

5. (Optional.) Specify the object list to be added to the notification triggered by the test.

   object list owner *group-owner* **name** *group-name*

   By default, no object list is specified for an existence trigger test.

6. Specify an existence trigger test type.

   type { **absent** | **changed** | **present** }

   The default existence trigger test types are **present** and **absent**.

7. Specify an existence trigger test type for the first sampling.

   startup { **absent** | **present** }

   By default, both the **present** and **absent** existence trigger test types are allowed for the first sampling.

Configuring a threshold trigger test

1. Enter system view.

   system-view

2. Enter trigger view.

   snmp mib event trigger owner *trigger-owner* **name** *trigger-name*

3. Specify a threshold test for the trigger and enter trigger-threshold view.

   test boolean

   By default ,no test is configured for a trigger.

4. Specify the object list to be added to the notification triggered by the test.

   object list owner *group-owner* **name** *group-name*

   By default, no object list is specified for a threshold trigger test.

5. (Optional.) Specify the type of the threshold trigger test for the first sampling.

   startup { **falling** | **rising** | **rising-or-falling** }

   The default threshold trigger test type for the first sampling is **rising-or-falling**.

6. Specify the delta falling threshold and the falling alarm event triggered when the delta value (difference between the current sampled value and the previous sampled value) is smaller than or equal to the delta falling threshold.

   delta falling { **event owner** *event-owner* **name** *event-name* | **value** *integer-value* }

   By default, the delta falling threshold is 0, and no falling alarm event is specified.

7. Specify the delta rising threshold and the rising alarm event triggered when the delta value is greater than or equal to the delta rising threshold.

   delta rising { **event owner** *event-owner* **name** *event-name* | **value** *integer-value* }

   By default, the delta rising threshold is 0, and no rising alarm event is specified.

8. Specify the falling threshold and the falling alarm event triggered when the sampled value is smaller than or equal to the threshold.

falling { **event owner** *event-owner* **name** *event-name* | **value** *integer-value* }

By default, the falling threshold is 0, and no falling alarm event is specified.

9. Specify the rising threshold and the ring alarm event triggered when the sampled value is greater than or equal to the threshold.

   rising { **event owner** *event-owner* **name** *event-name* | **value** *integer-value* }

   By default, the rising threshold is 0, and no rising alarm event is specified.

Enabling trigger sampling

Restrictions and guidelines

Enable trigger sampling after you complete trigger parameters configuration. You cannot modify trigger parameters after trigger sampling is enabled. To modify trigger parameters, first disable trigger sampling.

Procedure

1. Enter system view.

   system-view

2. Enter trigger view.

   snmp mib event trigger owner *trigger-owner* **name** *trigger-name*

3. Enable trigger sampling.

   trigger enable

   By default, trigger sampling is disabled.

Enabling SNMP notifications for the Event MIB module

About this task

To report critical Event MIB events to an NMS, enable SNMP notifications for the Event MIB module. For Event MIB event notifications to be sent correctly, you must also configure SNMP on the device. For more information about SNMP configuration, see the network management and monitoring configuration guide for the device.

Procedure

1. Enter system view.

   system-view

2. Enable snmp notifications for the Event MIB module.

   snmp-agent trap enable event-mib

   By default, SNMP notifications are enabled for the Event MIB module.

Display and maintenance commands for Event MIB

Execute **display** commands in any view.

Task	Command
Display Event MIB configuration and statistics.	**display snmp mib event**
Display event information.	**display snmp mib event event** [**owner** *event-owner* **name** *event-name*]
Display object list information.	**display snmp mib event object list** [**owner** *group-owner* **name** *group-name*]
Display global Event MIB configuration and statistics.	**display snmp mib event summary**
Display trigger information.	**display snmp mib event trigger** [**owner** *trigger-owner* **name** *trigger-name*]

Contents

Configuring process placement

About process placement

Process

A process contains a set of codes and provides specific functionality. For example, an AAA process provides AAA functions.

Each process runs in a protected memory space to prevent problems with one process from impacting the entire system.

Node

The term "node" in this document refers to CPUs that run the NF system. If the IRF fabric has multiple member devices, each member device has a node.

Process redundancy

The system backs up each active process running on one node to all the other nodes. When an active process fails, one of its standby processes promptly takes over without impacting any other service.

Process redundancy provides the following benefits:

- Improves service availability.
- Enables the system to quickly regain reliability after device status changes in such conditions as leave of an IRF member device.

Process placement

The execution of process placement policies varies by the location of active processes.

- An active process running only on the master device does not support placement optimization. If you configure a process placement policy for the process, the system displays a configuration failure message. When such an active process fails, the system automatically restarts the process. The standby processes are used for active/standby switchover and ISSU.
- Some active processes can run on either the master or subordinate device. When such an active process fails, the system uses a placement policy to select a new active process among standby processes.

Default process placement policy

The system provides a default process placement policy that takes effect for all processes. By default, the default process placement policy defines the following rules:

- The active process runs on the CPU of the master, and the standby processes run on the CPU of the subordinate device.
- A process runs at the location where it ran the last time and does not move to any other location during startup or operation.

- The addition of a new node does not impact current active processes. A new active process selects one node with sufficient CPU and memory resources. (You can use the **`display cpu-usage`** and **`display memory`** commands to view CPU and memory usage information.)

You can modify the default placement policy in the view you enter by using the **`placement program default`** command. You can also configure a placement policy for a specific process in the view you enter by using the **`placement program`** *`program-name`* `[` **`instance`** *`instance-name`* `]` command. A placement policy for a process takes precedence over the default process placement policy.

Process placement affinities

You can configure the following settings for a process placement policy to optimize process placement:

- **affinity location-set**—Location affinity, the preference for the process to run on a specific node.
- **affinity location-type**—Location type affinity, the preference for the process to run on a particular type of node. For more information about node types, see "Configuring a location type affinity."
- **affinity program**—Process affinity, the preference for the process to run on the same node as a particular process.
- **affinity self**—Self affinity, the preference for one instance of the process to run on the same node as any other instance of the process.

Affinities include positive affinities (**attract**) and negative affinities (**repulse**), all represented by integers in the range of 1 to 100000.

- The higher the attract value, the stronger the preference.
- The higher the repulse value, the weaker the preference.

Process placement optimization

After you apply new placement policies, the system makes placement decisions based on the new policies, node resources, and topology status. If the new location for an active process is different from the current node, the system changes the state of the process to standby, and uses the standby process on the preferred location as the new active process.

Restrictions: Hardware compatibility with process placement

Models	**Process placement compatibility**
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: process placement configuration

- Configuring process placement on an IRF fabric with only one member device does not change the location of processes. All processes run on the CPU of the IRF member device.
- Configuring process placement on an IRF fabric with multiple member devices places specific active processes to specific CPUs. In case of multiple CPUs, the system performs process redundancy. The number of standby processes and their CPU locations vary by function module. The system by default automatically determines the location for each active process, and process placement optimization is not required. If optimization is required, contact NSFOCUS Support to avoid service interruption.
- For an instance of a process, the priorities of the settings in placement policy view of an instance, placement policy view of a process, and the default placement policy view are in descending order. For a process, the settings in placement policy view of the process take precedence over the settings in the default placement policy view.
- To view the current location of an active process and its predicted new location after optimization, use the **display placement reoptimize** command.

Process placement tasks at a glance

To configure process placement, perform the following tasks:

1. Configuring process placement policy

   Choose the following tasks as needed:

   - Configuring a location affinity
   - Configuring a location type affinity
   - Configuring a process affinity
   - Configuring a self affinity
2. Optimizing process placement

Configuring process placement policy

Configuring a location affinity

1. Enter system view.

   system-view
2. Enter placement process view.
   - Enter default placement process view.

     placement program default
   - Enter specified placement process view.

     placement program { *program-name* [**instance** *instance-name*]

   Settings in default placement process view take effect for all processes. Settings in placement process view take effect only for the specified process.
3. Set the location affinity.

   affinity location-set { **slot** *slot-number* }&<1-5> { **attract** *strength* | **default** | **none** | **repulse** *strength* }

   By default, no location affinity is set.

Configuring a location type affinity

About this task

The following location types are available:

- **current**—Current location of the active process, which can be displayed with the **`display placement program`** command.
- **paired**—Locations of standby processes.
- **primary**—Master device.

Procedure

1. Enter system view.

   `system-view`

2. Enter placement process view.
   - Enter default placement process view.

     `placement program default`

   - Enter specified placement process view.

     `placement program` { *`program-name`* [**`instance`** *`instance-name`*]

   Settings in default placement process view take effect for all processes. Settings in placement process view take effect only for the specified process.

3. Set the location type affinity.

   `affinity location-type` { **`current`** | **`paired`** | **`primary`** } { **`attract`** *`strength`* | **`repulse`** *`strength`* | **`default`** | **`none`** }

   By default, no location type affinity is set.

Configuring a process affinity

1. Enter system view.

   `system-view`

2. Enter placement process view.
   - Enter default placement process view.

     `placement program default`

   - Enter specified placement process view.

     `placement program` { *`program-name`* [**`instance`** *`instance-name`*]

   Settings in default placement process view take effect for all processes. Settings in placement process view take effect only for the specified process.

3. Configure the affinity for the process to run on the same location as another process.

   `affinity program` *`program-name`* { **`attract`** *`strength`* | **`default`** | **`none`** | **`repulse`** *`strength`* }

   By default, no process affinity is set.

Configuring a self affinity

About this task

A self affinity determines the preference for one instance of a process to run on the same node as any other instance of the process. The self affinity setting does not take effect for a process that has only one instance.

Procedure

1. Enter system view.

   system-view

2. Enter placement process view.

   ○ Enter default placement process view.

   placement program default

   ○ Enter specified placement process view.

   placement program { *program-name* [**instance** *instance-name*]

   Settings in default placement process view take effect for all processes. Settings in placement process view take effect only for the specified process.

3. Configure a self affinity.

   affinity self { **attract** *strength* | **repulse** *strength* | **default** | **none** }

   By default, no self affinity is set.

Optimizing process placement

Restrictions and guidelines

To keep the system stable, do not perform any tasks that require process restart when you optimize process placement.

Procedure

1. Enter system view.

   system-view

2. Optimize process placement.

   placement reoptimize

△ **CAUTION:**

To avoid neighbor flapping of related protocols, make sure HA features such as NSR or GR are configured for the processes and are stable before optimizing process placement.

Display and maintenance commands for process placement

Execute **display** commands in any view.

Task	Command
Display service group information.	**display ha service-group** { *program-name* [**instance** *instance-name*] | **all** }
Display the running processes on a specific location.	**display placement location** { **all** | **slot** *slot-number* }
Display process placement policy information.	**display placement policy program** { program-name | **all** | **default** }

Task	Command
Display the location of a process.	**display placement program** { *program-name* | **all** }
Display the predicted location of a process after process placement optimization.	**display placement reoptimize program** { *program-name* [**instance** *instance-name*] | **all** }

NSFOCUS Firewall Series

NF VPN Instance Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for VPN instance features.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

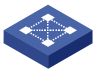
GUI conventions

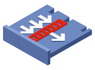
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring VPN instances

Basic concepts

Site

A site has the following features:

- A site is a group of IP systems with IP connectivity that does not rely on any service provider networks.
- The classification of a site depends on the topology relationship of the devices, rather than the geographical positions. However, the devices at a site are, in most cases, adjacent to each other geographically.
- The devices at a site can belong to multiple VPNs, which means that a site can belong to multiple VPNs.

Sites connected to the same provider network can be classified into different sets by policies. Only the sites in the same set can access each other through the provider network. Such a set is called a VPN.

VPN instance

In MPLS L3VPN, VPN instances implement route isolation for VPNs. A VPN instance is also called a Virtual Routing and Forwarding (VRF) instance.

Configuring VPN instances

Creating a VPN instance

A VPN instance is a collection of the VPN membership and routing rules of its associated site. A VPN instance might correspond to more than one VPN.

To create a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a VPN instance and enter its view.	**ip vpn-instance** *vpn-instance-name*	By default, no VPN instances exist.
3. (Optional.) Configure a description for the VPN instance.	**description** *text*	By default, no description is configured for a VPN instance.

Associating a VPN instance with an interface

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter interface view.	**interface** *interface-type interface-number*	N/A

Step	Command	Remarks
3. Associate a VPN instance with the interface.	**ip binding vpn-instance** *vpn-instance-name*	By default, an interface is not associated with a VPN instance and belongs to the public network. The **ip binding vpn-instance** command deletes the IP address of the interface. You must reconfigure an IP address for the interface after configuring the command.

Displaying and maintaining VPN instances

Execute **display** commands in any view.

Task	Command
Display VPN instance information.	**display ip vpn-instance** [**instance-name** *vpn-instance-name*]

NSFOCUS Firewall Series

NF VXLAN Instance Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for VXLAN instance features.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

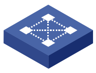
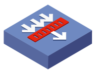
GUI conventions

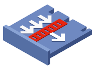
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
(!) **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

VXLAN overview

Virtual eXtensible LAN (VXLAN) is a MAC-in-UDP technology that provides Layer 2 connectivity between distant network sites across an IP network. VXLAN is typically used in data centers for multitenant services.

VXLAN benefits

VXLAN provides the following benefits:

- **Support for more virtual switched domains than VLANs**—Each VXLAN is uniquely identified by a 24-bit VXLAN ID. The total number of VXLANs can reach 16777216 (2^{24}). This specification makes VXLAN a better choice than 802.1Q VLAN to isolate traffic for user terminals.
- **Easy deployment and maintenance**—VXLAN requires deployment only on the edge devices of the transport network. Devices in the transport network perform typical Layer 3 forwarding.

VXLAN network model

As shown in Figure 1, a VXLAN is a virtual Layer 2 network (known as the overlay network) built on top of an existing physical Layer 3 network (known as the underlay network). The overlay network encapsulates inter-site Layer 2 frames into VXLAN packets and forwards the packets to the destination along the Layer 3 forwarding paths provided by the underlay network. The underlay network is transparent to tenants, and geographically dispersed sites of a tenant are merged into a Layer 2 network.

The transport edge devices assign user terminals to different VXLANs, and then forward traffic between sites for user terminals by using VXLAN tunnels. Supported user terminals include PCs, wireless terminals, and VMs on servers.

NOTE:

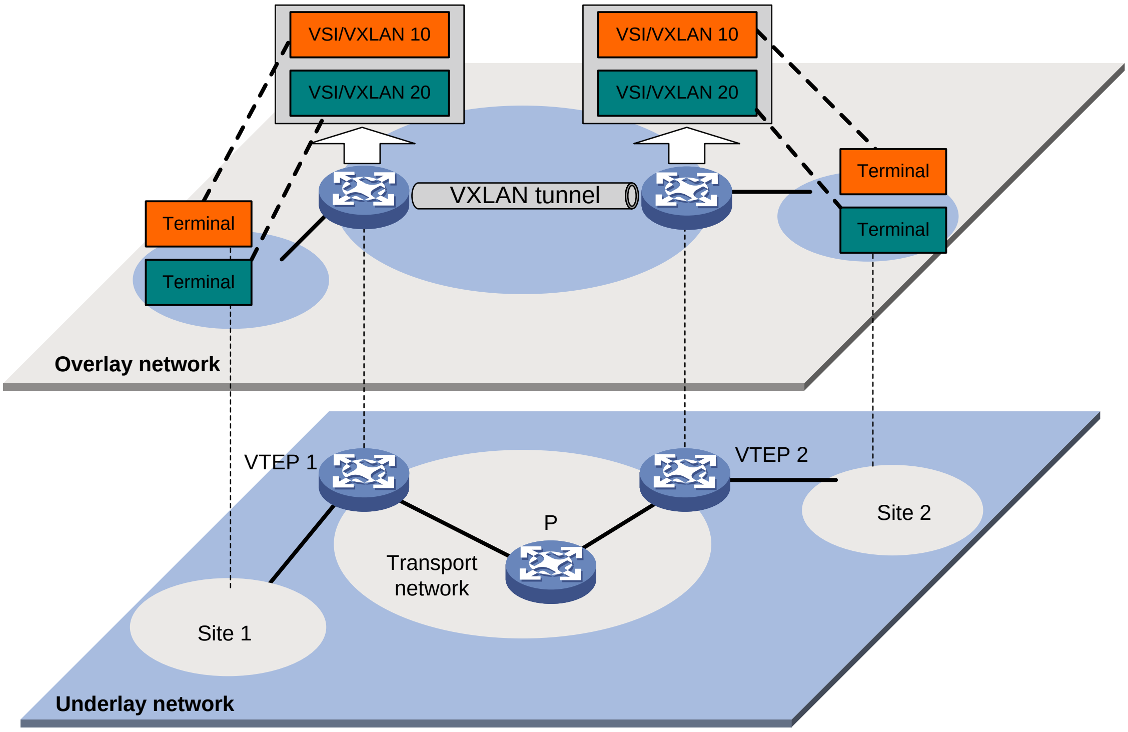
This document uses VMs as examples to describe the mechanisms of VXLAN. The mechanisms do not differ between different kinds of user terminals.

The transport edge devices are VXLAN tunnel endpoints (VTEP). The VTEP implementation of the device uses ACs, VSIs, and VXLAN tunnels to provide VXLAN services.

- **VSI**—A virtual switch instance is a virtual Layer 2 switched domain. Each VSI provides switching services only for one VXLAN. VSIs learn MAC addresses and forward frames independently of one another. VMs in different sites have Layer 2 connectivity if they are in the same VXLAN.
- **Attachment circuit (AC)**—An AC is a physical or virtual link that connects a VTEP to a local site. Typically, ACs are site-facing Layer 3 interfaces that are associated with the VSI of a VXLAN. Traffic received from an AC is assigned to the VSI associated with the AC.
- **VXLAN tunnel**—Logical point-to-point tunnels between VTEPs over the transport network. Each VXLAN tunnel can trunk multiple VXLANs.

VTEPs encapsulate VXLAN traffic in the VXLAN, outer UDP, and outer IP headers. The devices in the transport network forward VXLAN traffic only based on the outer IP header.

Figure 1 VXLAN network model

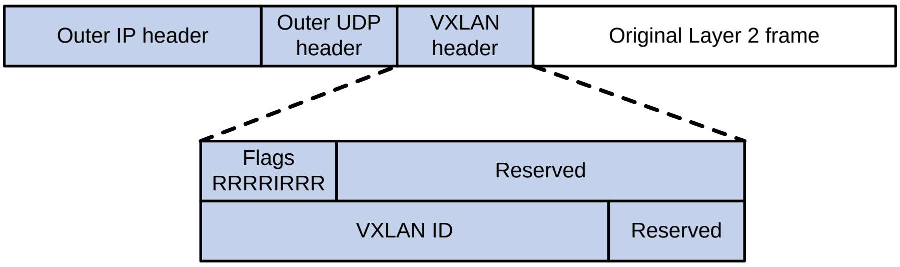


VXLAN packet format

As shown in Figure 2, a VTEP encapsulates a frame in the following headers:

- **8-byte VXLAN header**—VXLAN information for the frame.
  - **Flags**—If the I bit is 1, the VXLAN ID is valid. If the I bit is 0, the VXLAN ID is invalid. All other bits are reserved and set to 0.
  - **24-bit VXLAN ID**—Identifies the VXLAN of the frame. It is also called the virtual network identifier (VNI).
- **8-byte outer UDP header for VXLAN**—The default VXLAN destination UDP port number is 4789.
- **20-byte outer IPv4 or 40-byte outer IPv6 header**—Valid addresses of VTEPs or VXLAN multicast groups on the transport network. Devices in the transport network forward VXLAN packets based on the outer IP header.

Figure 2 VXLAN packet format



VXLAN working mechanisms

Generic VXLAN network establishment and forwarding process

The VTEP uses the following process to establish the VXLAN network and forward an inter-site frame:

1. Discovers remote VTEPs, establishes VXLAN tunnels, and assigns the VXLAN tunnels to VXLANs.
2. Assigns the frame to its matching VXLAN if the frame is sent between sites.
3. Performs MAC learning on the VXLAN's VSI.
4. Forwards the frame through VXLAN tunnels.

This section describes this process in detail. For intra-site frames in a VSI, the system performs typical Layer 2 forwarding, and it processes 802.1Q VLAN tags as described in "Access modes of VSIs."

VXLAN tunnel establishment and assignment

To provide Layer 2 connectivity for a VXLAN between two sites, you must create a VXLAN tunnel between the sites and assign the tunnel to the VXLAN.

VXLAN tunnel establishment

VXLAN supports manual VXLAN tunnel establishment. You must manually create a VXLAN tunnel interface, and specify the tunnel source and destination IP addresses on the peer VTEPs.

VXLAN tunnel assignment

VXLAN supports manual VXLAN tunnel assignment. You must manually assign VXLAN tunnels to VXLANs.

Assignment of traffic to VXLANs

Traffic from the local site to a remote site

The VTEP uses Layer 3 interface-to-VSI mapping to assign customer frames to a VXLAN. This method maps a site-facing Layer 3 interface to a VSI. The VTEP assigns all frames received from the interface to the VXLAN of the VSI.

Traffic from a remote site to the local site

When a frame arrives at a VXLAN tunnel, the VTEP uses the VXLAN ID in the frame to identify its VXLAN.

MAC learning

The VTEP performs source MAC learning on the VSI as a Layer 2 switch.

- For traffic from the local site to the remote site, the VTEP learns the source MAC address before VXLAN encapsulation.
- For traffic from the remote site to the local site, the VTEP learns the source MAC address after removing the VXLAN header.

A VSI's MAC address table includes the following types of MAC address entries:

- **Local MAC**—MAC entries dynamically learned from the local site. The outgoing interfaces for the MAC address entries are site-facing interfaces. VXLAN does not support static local-MAC entries.
- **Remote MAC**—MAC entries learned from a remote site. The outgoing interfaces for the MAC addresses are VXLAN tunnel interfaces.
  - **Static**—Manually added MAC entries.
  - **Dynamic**—MAC entries learned in the data plane from incoming traffic on VXLAN tunnels. The learned MAC addresses are contained in the inner Ethernet header.

  The following shows the priority order of different types of remote MAC address entries:

  a. Static MAC address entries.

  b. Dynamic MAC address entries.

Unicast forwarding

Intra-site unicast forwarding

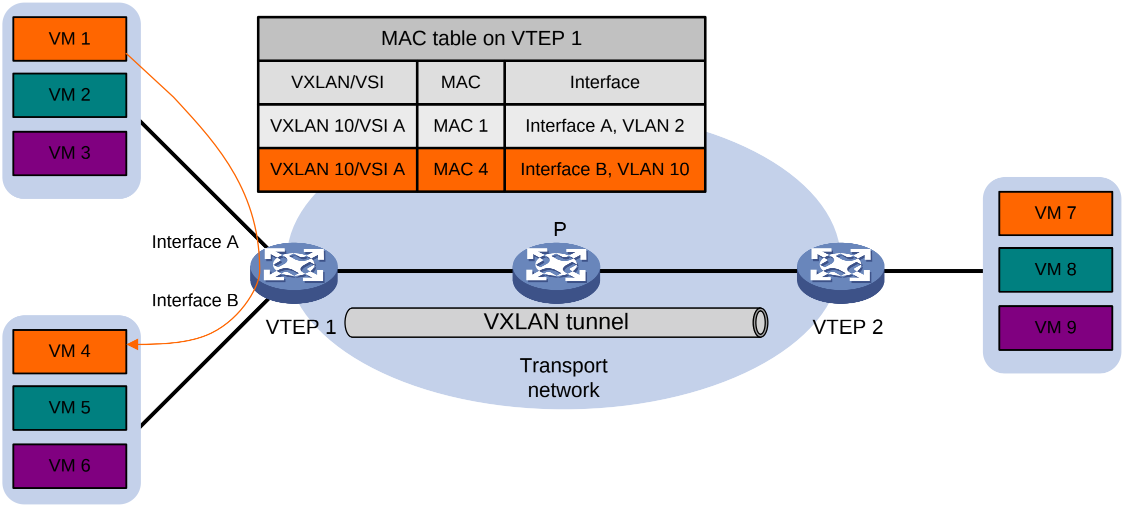
The VTEP uses the following process to forward a known unicast frame within a site:

1. Identifies the VSI of the frame.
2. Looks up the destination MAC address in the VSI's MAC address table for the outgoing interface.
3. Sends the frame out of the matching outgoing interface.

As shown in Figure 3, VTEP 1 forwards a frame from VM 1 to VM 4 within the local site in VLAN 10 as follows:

1. Identifies that the frame belongs to VSI A when the frame arrives at Interface A.
2. Looks up the destination MAC address (MAC 4) in the MAC address table of VSI A for the outgoing interface.
3. Sends the frame out of the matching outgoing interface (Interface B) to VM 4 in VLAN 10.

Figure 3 Intra-site unicast



MAC table on VTEP 1

VXLAN/VSI	MAC	Interface
VXLAN 10/VSI A	MAC 1	Interface A, VLAN 2
VXLAN 10/VSI A	MAC 4	Interface B, VLAN 10

Inter-site unicast forwarding

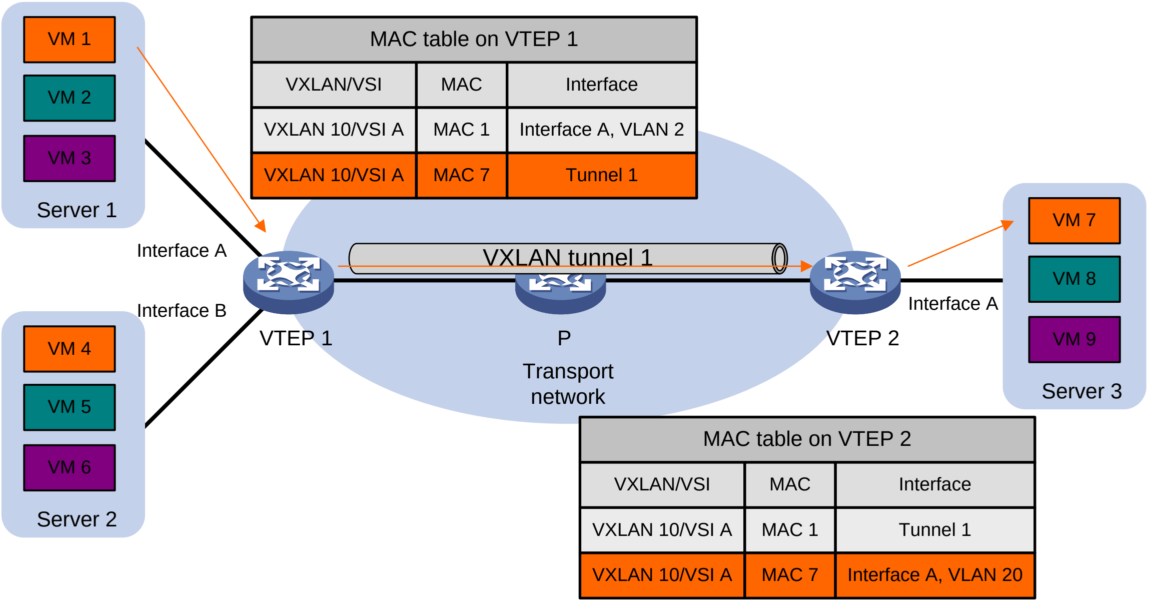
The following process (see Figure 4) applies to a known unicast frame between sites:

1. The source VTEP encapsulates the Ethernet frame in the VXLAN/UDP/IP header.

   In the outer IP header, the source IP address is the source VTEP's VXLAN tunnel source IP address. The destination IP address is the VXLAN tunnel destination IP address.

2. The source VTEP forwards the encapsulated packet out of the outgoing VXLAN tunnel interface found in the VSI's MAC address table.
3. The intermediate transport devices (P devices) forward the frame to the destination VTEP by using the outer IP header.
4. The destination VTEP removes the headers on top of the inner Ethernet frame. It then performs MAC address table lookup in the VXLAN's VSI to forward the frame out of the matching outgoing interface.

Figure 4 Inter-site unicast



MAC table on VTEP 1

VXLAN/VSI	MAC	Interface
VXLAN 10/VSI A	MAC 1	Interface A, VLAN 2
VXLAN 10/VSI A	MAC 7	Tunnel 1

MAC table on VTEP 2

VXLAN/VSI	MAC	Interface
VXLAN 10/VSI A	MAC 1	Tunnel 1
VXLAN 10/VSI A	MAC 7	Interface A, VLAN 20

Flood

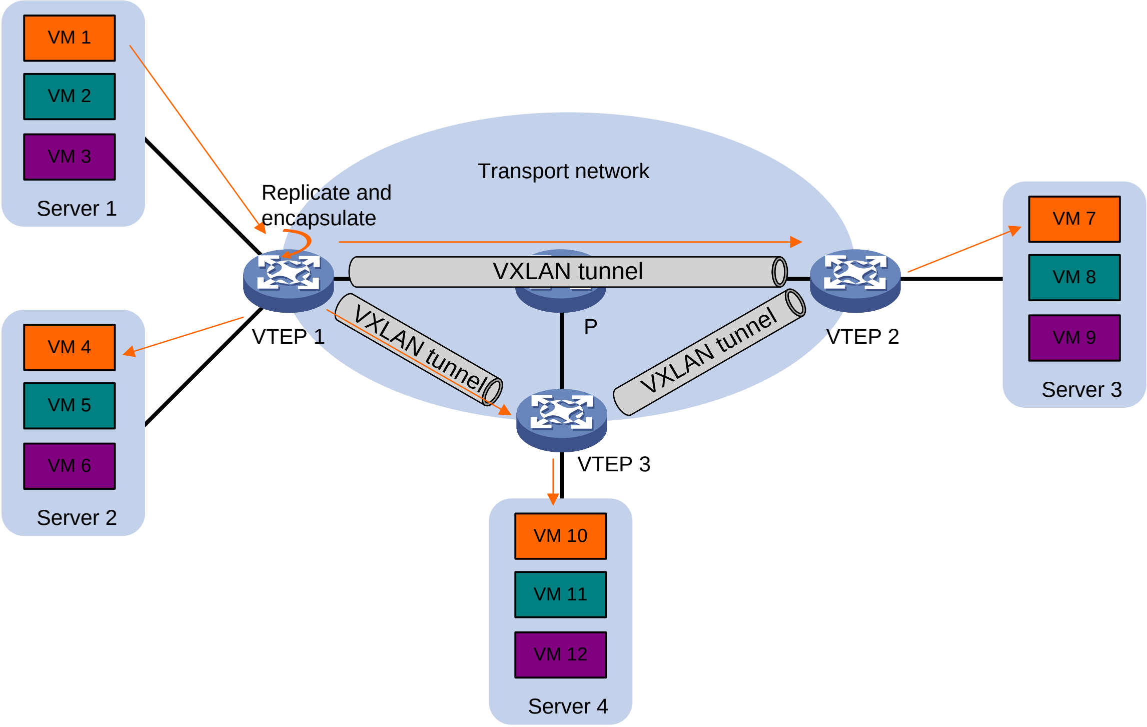
The source VTEP floods a broadcast, multicast, or unknown unicast frame to all site-facing interfaces and VXLAN tunnels in the VXLAN, except for the incoming interface. Each destination VTEP floods the inner Ethernet frame to all site-facing interfaces in the VXLAN. To avoid loops, the destination VTEPs do not flood the frame back to VXLAN tunnels.

VXLAN supports unicast mode (also called head-end replication) and flood proxy mode for flood traffic.

Unicast mode (head-end replication)

As shown in Figure 5, the source VTEP replicates the flood frame, and then sends one replica to the destination IP address of each VXLAN tunnel in the VXLAN.

Figure 5 Unicast mode

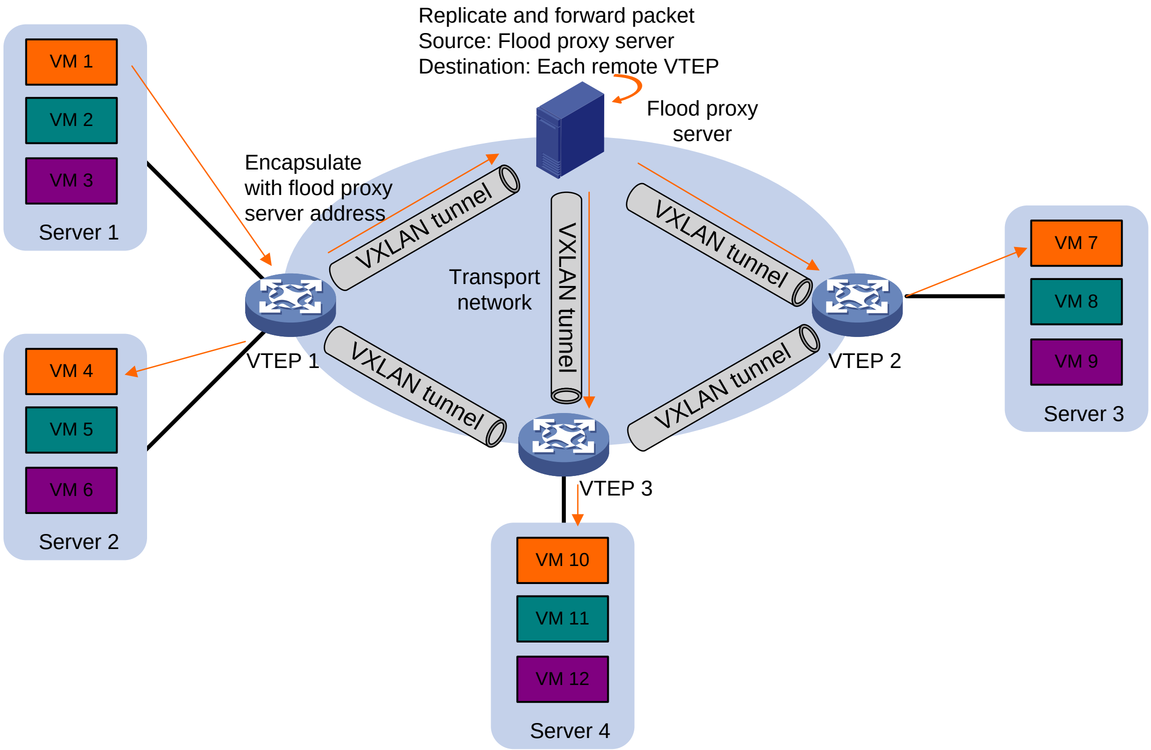


Flood proxy mode (proxy server replication)

As shown in Figure 6, the source VTEP sends the flood frame in a VXLAN packet over a VXLAN tunnel to a flood proxy server. The flood proxy server replicates and forwards the packet to each remote VTEP through its VXLAN tunnels.

The flood proxy mode applies to VXLANs that have many sites. This mode reduces flood traffic in the transport network without using a multicast protocol. To use a flood proxy server, you must set up a VXLAN tunnel to the server on each VTEP.

Figure 6 Flood proxy mode



The flood proxy mode is typically used in SDN transport networks that have a virtual server as the flood proxy server. For VTEPs to forward packets based on the MAC address table issued by an SDN controller, you must perform the following tasks on the VTEPs:

- Disable remote-MAC address learning by using the **`vxlan tunnel mac-learning disable`** command.
- Disable source MAC check on all transport-facing interfaces by using the **`undo mac-address static source-check enable`** command. If the VTEP is an IRF fabric, you must also disable the feature on all IRF ports.

Access modes of VSIs

The access mode of a VSI determines how the VTEP processes the 802.1Q VLAN tags in the Ethernet frames.

VLAN access mode

In this mode, Ethernet frames received from or sent to the local site must contain 802.1Q VLAN tags.

- For an Ethernet frame received from the local site, the VTEP removes all its 802.1Q VLAN tags before forwarding the frame.
- For an Ethernet frame destined for the local site, the VTEP adds 802.1Q VLAN tags to the frame before forwarding the frame.

In VLAN access mode, VXLAN packets sent between sites do not contain 802.1Q VLAN tags. You can use different 802.1Q VLANs to provide the same service in different sites.

Ethernet access mode

The VTEP does not process the 802.1Q VLAN tags of Ethernet frames received from or sent to the local site.

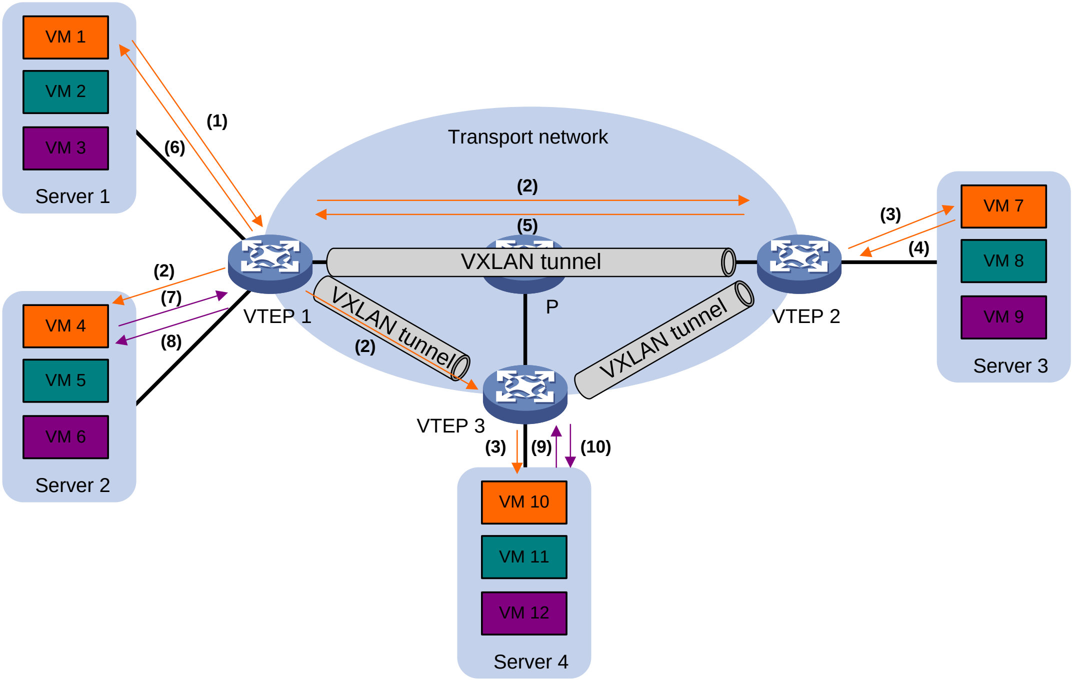
- For an Ethernet frame received from the local site, the VTEP forwards the frame with the 802.1Q VLAN tags intact.
- For an Ethernet frame destined for the local site, the VTEP forwards the frame without adding 802.1Q VLAN tags.

In Ethernet access mode, VXLAN packets sent between VXLAN sites contain 802.1Q VLAN tags. You must use the same VLAN to provide the same service between sites.

ARP flood suppression reduces ARP request broadcasts by enabling the VTEP to reply to ARP requests on behalf of VMs.

As shown in Figure 7, this feature snoops ARP packets to populate the ARP flood suppression table with local and remote MAC addresses. If an ARP request has a matching entry, the VTEP replies to the request on behalf of the VM. If no match is found, the VTEP floods the request to both local and remote sites.

Figure 7 ARP flood suppression



ARP flood suppression uses the following workflow:

1. VM 1 sends an ARP request to obtain the MAC address of VM 7.
2. VTEP 1 creates a suppression entry for VM 1, and floods the ARP request in the VXLAN.
3. VTEP 2 and VTEP 3 de-encapsulate the ARP request. The VTEPs create a suppression entry for VM 1, and broadcast the request in the local site.
4. VM 7 sends an ARP reply.
5. VTEP 2 creates a suppression entry for VM 7 and forwards the ARP reply to VTEP 1.
6. VTEP 1 de-encapsulates the ARP reply, creates a suppression entry for VM 7, and forwards the ARP reply to VM 1.
7. VM 4 sends an ARP request to obtain the MAC address of VM 1 or VM 7.
8. VTEP 1 creates a suppression entry for VM 4 and replies to the ARP request.
9. VM 10 sends an ARP request to obtain the MAC address of VM 1.
10. VTEP 3 creates a suppression entry for VM 10 and replies to the ARP request.

VXLAN IP gateways

A VXLAN IP gateway provides Layer 3 forwarding services for VMs in VXLANs. A VXLAN IP gateway can be an independent device or be collocated with a VTEP. For more information about VXLAN IP gateway placement, see "Configuring VXLAN IP gateways."

Protocols and standards

RFC 7348, *Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks*

Configuring basic VXLAN features

VXLAN tasks at a glance

To configure basic VXLAN settings, perform the following tasks on VTEPs:

1. Setting the forwarding mode for VXLANs
2. Creating a VXLAN on a VSI
3. Configuring a VXLAN tunnel
4. Manually assigning VXLAN tunnels to a VXLAN
5. Assigning customer frames to a VSI
6. (Optional.) Managing MAC address entries
   - Configuring static remote-MAC address entries
   - Disabling remote-MAC address learning
   - Enabling local-MAC logging
7. (Optional.) Configuring VXLAN packet parameters
   - Setting the destination UDP port number of VXLAN packets
   - Setting the source UDP port number of VXLAN packets
   - Setting a service class value for outgoing VXLAN packets
   - Configuring VXLAN packet check
8. (Optional.) Reducing flood traffic in the transport network
   - Confining unknown-unicast floods to the local site
   - Enabling ARP flood suppression
9. (Optional.) Enabling VXLAN packet statistics
10. (Optional.) Enabling VXLAN fast forwarding

Prerequisites for VXLAN

Configure a routing protocol on the devices in the transport network to make sure the VTEPs can reach one another.

Setting the forwarding mode for VXLANs

About this task

A VXLAN tunnel supports the following modes:

- **Layer 3 forwarding mode**—The device uses the ARP table (IPv4 network) or ND table (IPv6 network) to forward traffic for VXLANs.
- **Layer 2 forwarding mode**—The device uses the MAC address table to forward traffic for VXLANs.

If the device is a VTEP, enable Layer 2 forwarding for VXLANs. If the device is a VXLAN IP gateway, enable Layer 3 forwarding for VXLANs. For more information about VXLAN IP gateways, see "Configuring VXLAN IP gateways."

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

You must delete all VSIs, VSI interfaces, and VXLAN tunnel interfaces before you can change the forwarding mode. As a best practice, finish VXLAN network planning and determine the VXLAN forwarding mode of each device before your configuration, and set the VXLAN forwarding mode before other VXLAN settings.

Procedure

1. Enter system view.

   `system-view`

2. Set the forwarding mode of VXLANs.
   - Enable Layer 2 forwarding.

     `undo vxlan ip-forwarding`
   - Enable Layer 3 forwarding.

     `vxlan ip-forwarding`

   By default, Layer 3 forwarding is enabled for VXLANs.

Creating a VXLAN on a VSI

1. Enter system view.

   `system-view`

2. Enable L2VPN.

   `l2vpn enable`

   By default, L2VPN is disabled.

3. Create a VSI and enter VSI view.

   `vsi` *`vsi-name`*

4. Enable the VSI.

   `undo shutdown`

   By default, a VSI is enabled.

5. Create a VXLAN and enter VXLAN view.

   `vxlan` *`vxlan-id`*

   You can create only one VXLAN on a VSI.

   The VXLAN ID must be unique for each VSI.

6. (Optional.) Configure VSI parameters:

   a. Return to VSI view.

      `quit`

   b. Configure a VSI description.

      `description` *`text`*

      By default, a VSI does not have a description.

c. Set the MTU for the VSI.

```
mtu mtu
```

The default MTU for a VSI is 1500 bytes.

The MTU set by using this command limits the maximum length of the packets that a VSI receives from ACs and forwards through VXLAN tunnels. The MTU does not limit the maximum length of other packets in the VXLAN VSI.

d. Enable MAC address learning for the VSI.

```
mac-learning enable
```

By default, MAC address learning is enabled for a VSI.

e. Set a limit for the VSI's MAC address table.

```
mac-table limit mac-limit
```

By default, no limit is set for a VSI's MAC address table.

Configuring a VXLAN tunnel

Manually creating a VXLAN tunnel

About this task

When you manually create a VXLAN tunnel, specify addresses on the local VTEP and the remote VTEP as the tunnel source and destination addresses, respectively.

Restrictions and guidelines

As a best practice, do not configure multiple VXLAN tunnels to use the same source and destination IP addresses.

This task provides basic VXLAN tunnel configuration. For more information about tunnel configuration and commands, see *Layer 3—IP Services Configuration Guide* and *Layer 3—IP Services Command Reference*.

Procedure

1. Enter system view.

   ```
   system-view
   ```

   A VXLAN tunnel uses the global source address if you do not specify a source interface or source address for the tunnel.

2. Create a VXLAN tunnel interface and enter tunnel interface view.

   ```
   interface tunnel tunnel-number mode vxlan
   ```

   The endpoints of a tunnel must use the same tunnel mode.

3. Specify a source address for the tunnel. Choose one of the following methods:

   - Specify a source IP address for the tunnel.

     ```
     source { ipv4-address | ipv6-address }
     ```

     The specified IP address is used in the outer IP header of tunneled VXLAN packets.

   - Specify a source interface for the tunnel.

     ```
     source interface-type interface-number
     ```

     The primary IP address of the specified interface is used in the outer IP header of tunneled VXLAN packets.

   By default, no source IP address or source interface is specified for a tunnel.

4. Specify a destination IP address for the tunnel.

   ```
   destination { ipv4-address | ipv6-address }
   ```

By default, no destination IP address is specified for a tunnel.

Specify the remote VTEP's IP address. This IP address will be the destination IP address in the outer IP header of tunneled VXLAN packets.

Enabling BFD on a VXLAN tunnel

About this task

Enable BFD on both ends of a VXLAN tunnel for quick link connectivity detection. The VTEPs periodically send BFD single-hop control packets to each other through the VXLAN tunnel. A VTEP sets the tunnel state to Defect if it has not received control packets from the remote end for 5 seconds. In this situation, the tunnel interface state is still Up. The tunnel state will change from Defect to Up if the VTEP can receive BFD control packets again.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

You must enable BFD on both ends of a VXLAN tunnel.

Procedure

1. Enter system view.

   `system-view`

2. Specify the reserved VXLAN.

   `reserved vxlan` *`vxlan-id`*

   By default, no VXLAN has been reserved.

   For BFD sessions to come up, you must reserve a VXLAN.

   You can specify only one reserved VXLAN on the VTEP. The reserved VXLAN cannot be the VXLAN created on any VSI.

   The reserved VXLAN ID cannot be the same as any remote VXLAN ID specified by using the **`mapping vni`** command. For more information about the **`mapping vni`** command, see *EVPN Command Reference*.

3. Enter VXLAN tunnel interface view.

   `interface tunnel` *`tunnel-number`*

4. Enable BFD on the tunnel.

   `tunnel bfd enable destination-mac` *`mac-address`*

   By default, BFD is disabled on a tunnel.

Manually assigning VXLAN tunnels to a VXLAN

About this task

To provide Layer 2 connectivity for a VXLAN between two sites, you must assign the VXLAN tunnel between the sites to the VXLAN.

You can assign multiple VXLAN tunnels to a VXLAN, and configure a VXLAN tunnel to trunk multiple VXLANs. For a unicast-mode VXLAN, the system floods unknown unicast, multicast, and broadcast traffic to each tunnel associated with the VXLAN.

Restrictions and guidelines

For full Layer 2 connectivity in the VXLAN, make sure the VXLAN contains the VXLAN tunnel between each pair of sites in the VXLAN.

Procedure

1. Enter system view.

   `system-view`

2. Enter VSI view.

   `vsi` *`vsi-name`*

3. Enter VXLAN view.

   `vxlan` *`vxlan-id`*

4. Assign VXLAN tunnels to the VXLAN.

   `tunnel` *`tunnel-number`*

   By default, a VXLAN does not contain any VXLAN tunnels.

Assigning customer frames to a VSI

Mapping a Layer 3 interface to a VSI

About this task

To assign the customer traffic on a Layer 3 interface to a VXLAN, map the interface to the VXLAN's VSI. The VSI uses its MAC address table to forward the customer traffic.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Procedure

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Map the Layer 3 interface to a VSI.

   `xconnect vsi` *`vsi-name`* `[` **`track`** *`track-entry-number`*`&<1-3> ]`

   By default, a Layer 3 interface is not mapped to any VSI.

   If the AC is a Layer 3 subinterface, you can specify the access mode. The default access mode is VLAN. If the AC is a Layer 3 interface, you cannot specify the access mode.

Mapping an Ethernet service instance to a VSI

About this task

An Ethernet service instance matches a list of VLANs on a site-facing interface. The VTEP assigns customer traffic from the VLANs to a VXLAN by mapping the Ethernet service instance to a VSI.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

An Ethernet service instance can contain only one match criterion. To change the match criterion, you must remove the original criterion first. When you remove the match criterion in an Ethernet service instance, the mapping between the service instance and the VSI is removed automatically.

Procedure

1. Enter system view.

   system-view

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     interface *interface-type interface-number*
   - Enter Layer 2 aggregate interface view.

     interface bridge-aggregation *interface-number*

3. Create an Ethernet service instance and enter Ethernet service instance view.

   service-instance *instance-id*

4. Configure a frame match criterion. Choose one of the following options:
   - Match frames tagged with the specified inner 802.1Q VLAN IDs.

     encapsulation c-vid *vlan-id-list*
   - Match frames tagged with the specified outer 802.1Q VLAN IDs.

     encapsulation s-vid *vlan-id-list* [**only-tagged**]
   - Match frames tagged with the specified outer and inner 802.1Q VLAN IDs.

     encapsulation s-vid *vlan-id* **c-vid** { *vlan-id-list* | **all** }
   - Match any 802.1Q tagged or untagged frames.

     encapsulation { **tagged** | **untagged** }
   - Match frames that do not match any other service instance on the interface.

     encapsulation default

     An interface can contain only one Ethernet service instance that uses the **encapsulation default** match criterion.

     An Ethernet service instance that uses the **encapsulation default** match criterion matches any frames if it is the only instance on the interface.

   By default, an Ethernet service instance does not contain a frame match criterion.

5. Map the Ethernet service instance to a VSI.

xconnect vsi *vsi-name* [**access-mode** { **ethernet** | **vlan** }] [**track** *track-entry-number*&<1-3>]

By default, an Ethernet service instance is not mapped to any VSI.

Managing MAC address entries

About MAC address entry management

Local-MAC address entries are only learned dynamically. You can log local MAC addresses and local-MAC changes.

Remote-MAC address entries have a variety of types, including manually added entries and dynamically learned entries.

Configuring static remote-MAC address entries

1. Enter system view.

   system-view

2. Add a static remote-MAC address entry.

   mac-address static *mac-address* **interface tunnel** *tunnel-number* **vsi** *vsi-name*

   For the setting to take effect, make sure the VSI's VXLAN has been specified on the VXLAN tunnel.

Disabling remote-MAC address learning

About this task

When network attacks occur, disable remote-MAC address learning to prevent the device from learning incorrect remote MAC addresses. You can manually add static remote-MAC address entries.

Procedure

1. Enter system view.

   system-view

2. Disable remote-MAC address learning.

   vxlan tunnel mac-learning disable

   By default, remote-MAC address learning is enabled.

Enabling local-MAC logging

About this task

When the local-MAC logging feature is enabled, the VXLAN module immediately sends a log message with its local MAC addresses to the information center. When a local MAC address is added or removed, a log message is also sent to the information center to notify the local-MAC change.

With the information center, you can set log message filtering and output rules, including output destinations. For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

Procedure

1. Enter system view.

   system-view

2. Enable local-MAC logging.

   vxlan local-mac report

   By default, local-MAC logging is disabled.

Setting the destination UDP port number of VXLAN packets

1. Enter system view.

   system-view

2. Set a destination UDP port for VXLAN packets.

   vxlan udp-port *port-number*

   By default, the destination UDP port number is 4789 for VXLAN packets.

   You must configure the same destination UDP port number on all VTEPs in a VXLAN.

Setting the source UDP port number of VXLAN packets

About this task

Perform this task to enable a VXLAN tunnel interface to encapsulate different source UDP port numbers for traffic flows. This allows IPsec to identify the VXLAN packets to encrypt by the source UDP port number in the VXLAN encapsulation.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines

The commands used for setting the source UDP port number of VXLAN packets take effect only on IPv4-based VXLAN. Only manually created VXLAN tunnel interfaces support these commands.

The **vxlan source udp-port acl** command has a higher priority than the **vxlan source udp-port five-tuple** command. If you use both commands on a VXLAN tunnel interface, the **vxlan source udp-port five-tuple** command takes effect only on the frames that fail to match the ACL specified by using the **vxlan source udp-port acl** command.

Procedure

1. Enter system view.

   system-view

2. Enter VXLAN tunnel interface view.

   interface tunnel *tunnel-number* **mode vxlan** [**ipv6**]

3. Set a source UDP port for VXLAN packets.
   - Configure an ACL match criterion and specify the source UDP port number in the VXLAN encapsulation for matching frames.

     ```
     vxlan source udp-port port-number acl acl-number
     ```

     If you execute this command multiple times, the most recent configuration takes effect.

     If the ACL specified by using this command does not exist or does not contain an IP address-related rule, frames are encapsulated based on the default setting.
   - Generate the source UDP port number in the VXLAN encapsulation based on the IP five-tuple of the inner Ethernet frame.

     ```
     vxlan source udp-port five-tuple
     ```

   By default, the source UDP port number in the VXLAN encapsulation is generated based on the source and destination MAC addresses of the inner Ethernet frame.

Setting a service class value for outgoing VXLAN packets

About this task

Class Based Tunnel Selection (CBTS) compares the service class value of VXLAN packets with the service class values of MPLS TE tunnels. CBTS uses the following rules to select a tunnel to forward the traffic:

- If the packets match only one MPLS TE tunnel, CBTS uses this tunnel.
- If the packets match multiple MPLS TE tunnels, CBTS randomly selects one tunnel from them.
- If the packets do not match any MPLS TE tunnel, CBTS selects the MPLS TE tunnel that meets the following requirements:
  - Its service class value is smaller than the service class value of the traffic.
  - Its service class value is the nearest to the service class value of the traffic.

  If multiple qualified tunnels exist, CBTS randomly selects one of them to forward the packets. If an MPLS TE tunnel is not configured with a service class value, this tunnel has the smallest service class value.

The service class value is used only when MPLS TE tunnels are used to forward the VXLAN packets. This value is meaningless if any other tunnel is used to forward the VXLAN packets.

To set the service class value for an MPLS TE tunnel, use the **`mpls te service-class`** command. For more information about this command, see MPLS TE commands in *MPLS Command Reference*.

Hardware and feature compatibility

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter VSI view.

```
vsi vsi-name
```

3. Set a service class value for outgoing VXLAN packets.

   ```
   service-class service-class-value
   ```

   By default, no service class value is set for outgoing VXLAN packets.

Configuring VXLAN packet check

About this task

The device always sets the UDP checksum of VXLAN packets to zero. For compatibility with third-party devices, a VXLAN packet can pass the check if its UDP checksum is zero or correct. If its UDP checksum is incorrect, the VXLAN packet fails the check and is dropped.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enable the VTEP to drop VXLAN packets that fail UDP checksum check.

   ```
   vxlan invalid-udp-checksum discard
   ```

   By default, the VTEP does not check the UDP checksum of VXLAN packets.

Confining unknown-unicast floods to the local site

About this task

By default, the VTEP floods unknown unicast frames received from the local site to the following interfaces in the frame's VXLAN:

- All site-facing interfaces except for the incoming interface.
- All VXLAN tunnel interfaces.

To exclude a remote MAC address from the flood suppression done by using this feature, enable selective flood for the MAC address. The VTEP will flood the frames destined for the MAC address to remote sites.

Procedure

1. Enter system view.

   ```
   system-view
   ```

2. Enter VSI view.

   ```
   vsi vsi-name
   ```

3. Disable the VSI to flood unknown unicast traffic to VXLAN tunnel interfaces.

   ```
   flooding disable
   ```

   By default, unknown unicast traffic is flooded to all interfaces in the VXLAN, except for the incoming interface.

4. (Optional.) Enable selective flood for a MAC address.

   ```
   selective-flooding mac-address mac-address
   ```

Enabling ARP flood suppression

Restrictions and guidelines

The aging timer is fixed at 25 minutes for ARP flood suppression entries. If the suppression table is full, the VTEP stops learning new entries. For the VTEP to learn new entries, you must wait for old entries to age out, or use the **reset arp suppression vsi** command to clear the table.

If the **flooding disable** command is configured, set the MAC aging timer to a higher value than the aging timer for ARP flood suppression entries on all VTEPs. This setting prevents the traffic blackhole that occurs when a MAC address entry ages out before its ARP flood suppression entry ages out. To set the MAC aging timer, use the **mac-address timer** command.

When remote ARP learning is disabled for VXLANs, the device does not use ARP flood suppression entries to respond to ARP requests received on VXLAN tunnels.

Procedure

1. Enter system view.

   system-view

2. Enter VSI view.

   vsi *vsi-name*

3. Enable ARP flood suppression.

   arp suppression enable

   By default, ARP flood suppression is disabled.

Enabling VXLAN packet statistics

Hardware compatibility with VXLAN packet statistics

Models	Feature compatibility
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Enabling packet statistics for a VSI

Restrictions and guidelines

To display the packet statistics for a VSI, use the **display l2vpn vsi verbose** command in any view.

To clear the packet statistics for a VSI, use the **reset l2vpn statistics vsi** command in user view.

Procedure

1. Enter system view.

   system-view

2. Enter VSI view.

   vsi *vsi-name*

3. Enable packet statistics for the VSI.

   `statistics enable`

   By default, the packet statistics feature is disabled for all VSIs.

Enabling packet statistics for an AC

Restrictions and guidelines

For the **`ac statistics enable`** command to take effect on a Layer 3 interface, you must map the Layer 3 interface to a VSI. When you modify the VSI mapping, the packet statistics of the interface are cleared.

For the **`statistics enable`** command to take effect on an Ethernet service instance, you must configure a frame match criterion for the Ethernet service instance and map it to a VSI. When you modify the frame match criterion or VSI mapping, the packet statistics of the instance are cleared.

Enabling packet statistics for a Layer 3 interface

1. Enter system view.

   `system-view`

2. Enter Layer 3 interface view.

   `interface` *`interface-type interface-number`*

3. Enable packet statistics for the Layer 3 interface.

   `ac statistics enable`

   By default, the packet statistics feature is disabled for Layer 3 interfaces that act as ACs.

Enabling packet statistics for an Ethernet service instance

1. Enter system view.

   `system-view`

2. Enter interface view.
   - Enter Layer 2 Ethernet interface view.

     `interface` *`interface-type interface-number`*
   - Enter Layer 2 aggregate interface view.

     `interface bridge-aggregation` *`interface-number`*

3. Enter Ethernet service instance view.

   `service-instance` *`instance-id`*

4. Enable packet statistics for the Ethernet service instance.

   `statistics enable`

   By default, the packet statistics feature is disabled for all Ethernet service instances.

Enable packet statistics for all VXLAN tunnels of a VSI

About this task

VXLAN tunnels can be manually or automatically created. You can enable packet statistics for all VXLAN tunnels of a VSI.

If you enable packet statistics in VSI view, follow these guidelines:

- To display the packet statistics for VXLAN tunnels, use the **`display vxlan tunnel`** command in any view.

- To clear the packet statistics for VXLAN tunnels, use the **`reset l2vpn statistics tunnel`** command in user view.

Procedure

1. Enter system view.

   `system-view`

2. Enter VSI view.

   `vsi` *`vsi-name`*

3. Enable packet statistics for all VXLAN tunnels associated with the VSI.

   `tunnel statistics enable`

   By default, the packet statistics feature is disabled for the VXLAN tunnels associated with a VSI.

   This command enables packet statistics only for VXLAN tunnels. It does not take effect on VXLAN-DCI tunnels.

Setting the VXLAN statistics collection interval

1. Enter system view.

   `system-view`

2. Set the VXLAN statistics collection interval.

   `l2vpn statistics interval` *`interval`*

   By default, the VXLAN statistics collection interval is 15 minutes.

Enabling VXLAN fast forwarding

About this task

VXLAN fast forwarding enables the device to bypass QoS and security services when the device forwards data traffic over VXLAN tunnels based on the software. As a best practice, enable this feature to improve forwarding speed only when QoS and security services are not configured on the following interfaces:

- VSI interfaces.
- Traffic outgoing interfaces for VXLAN tunnels.

Restrictions and guidelines

When VXLAN fast forwarding is enabled, a VXLAN tunnel cannot use ECMP routes to load share traffic. Instead, it selects one route from the ECMP routes to forward VXLAN packets.

Procedure

1. Enter system view.

   `system-view`

2. Enable VXLAN fast forwarding.

   `vxlan fast-forwarding enable`

   By default, VXLAN fast forwarding is disabled.

Display and maintenance commands for VXLANs

Execute **`display`** commands in any view and **`reset`** commands in user view.

IMPORTANT:

For support of hardware platforms for the display and maintenance commands, see the command reference.

Task	Command
Display ARP flood suppression entries on VSIs.	**display arp suppression vsi** [**name** *vsi-name*] [**slot** *slot-number*] [**count**]
Display information about tunnel interfaces.	**display interface** [**tunnel** [*number*]] [**brief** [**description** | **down**]]
Display L2VPN information for Layer 3 interfaces that are mapped to VSIs.	**display l2vpn interface** [**vsi** *vsi-name* | *interface-type interface-number*] [**verbose**]
Display MAC address entries for VSIs.	**display l2vpn mac-address** [**vsi** *vsi-name*] [**dynamic**] [**count**]
Display information about Ethernet service instances.	**display l2vpn service-instance** [**interface** *interface-type interface-number* [**service-instance** *instance-id*]] [**verbose**]
Display information about VSIs.	**display l2vpn vsi** [**name** *vsi-name*] [**verbose**]
Display VXLAN tunnel information for VXLANs.	**display vxlan tunnel** [**vxlan** *vxlan-id* [**tunnel** *tunnel-number*]]
Clear ARP flood suppression entries on VSIs.	**reset arp suppression vsi** [**name** *vsi-name*]
Clear dynamic MAC address entries on VSIs.	**reset l2vpn mac-address** [**vsi** *vsi-name*]
Clear packet statistics on ACs.	**reset l2vpn statistics ac** [**interface** *interface-type interface-number* [**service-instance** *instance-id*]]
Clear packet statistics on VXLAN tunnel interfaces.	**reset l2vpn statistics tunnel** [**vsi** *vsi-name*]
Clear packet statistics on VSIs.	**reset l2vpn statistics vsi** [**name** *vsi-name*]

NOTE:

For more information about the **display interface tunnel** command, see tunneling commands in *Layer 3—IP Services Command Reference*.

Configuring VXLAN IP gateways

About VXLAN IP gateways

The following are available IP gateway placement designs for VXLANs:

- **VXLAN IP gateways separated from VTEPs**—Use a VXLAN-unaware device as a gateway to the external network for VXLANs. On the gateway, you do not need to configure VXLAN settings.
- **VXLAN IP gateways collocated with VTEPs**—Include the following placement designs:
  - **Centralized VXLAN IP gateway deployment**—Use one VTEP to provide Layer 3 forwarding for VXLANs. Typically, the gateway-collocated VTEP connects to other VTEPs and the external network. To use this design, make sure the IP gateway has sufficient bandwidth and processing capability. Centralized VXLAN IP gateways provide services only for IPv4 networks.
  - **Centralized VXLAN gateway group deployment**—Use one VTEP group that contains redundant centralized VXLAN IP gateways to provide reliable gateway services for VXLANs.
  - **Distributed VXLAN IP gateway deployment**—Deploy one VXLAN IP gateway on each VTEP to provide Layer 3 forwarding for VXLANs at their respective sites. This design distributes the Layer 3 traffic load across VTEPs. However, its configuration is more complex than the centralized VXLAN IP gateway design. Distributed gateways can provide services for both IPv4 and IPv6 networks.

In a collocation design, the VTEPs use virtual Layer 3 VSI interfaces as gateway interfaces to provide services for VXLANs.

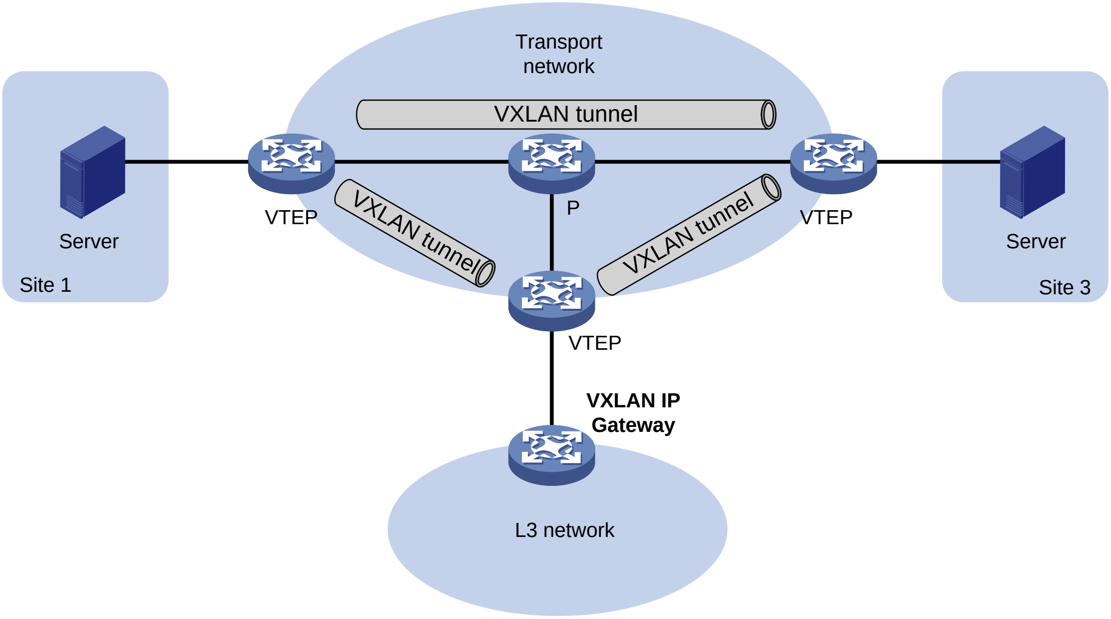
NOTE:

The following information describes traffic forwarding of VXLAN IP gateways in IPv4 networks. Traffic forwarding of VXLAN IP gateways in IPv6 networks is similar to that in IPv4 networks.

VXLAN IP gateways separated from VTEPs

As shown in Figure 8, an independent VXLAN IP gateway connects a Layer 3 network to a VTEP. VMs send Layer 3 traffic in Layer 2 frames to the gateway through VXLAN tunnels. When the tunneled VXLAN packets arrive, the VTEP terminates the VXLANs and forwards the inner frames to the gateway. In this gateway placement design, the VTEP does not perform Layer 3 forwarding for VXLANs.

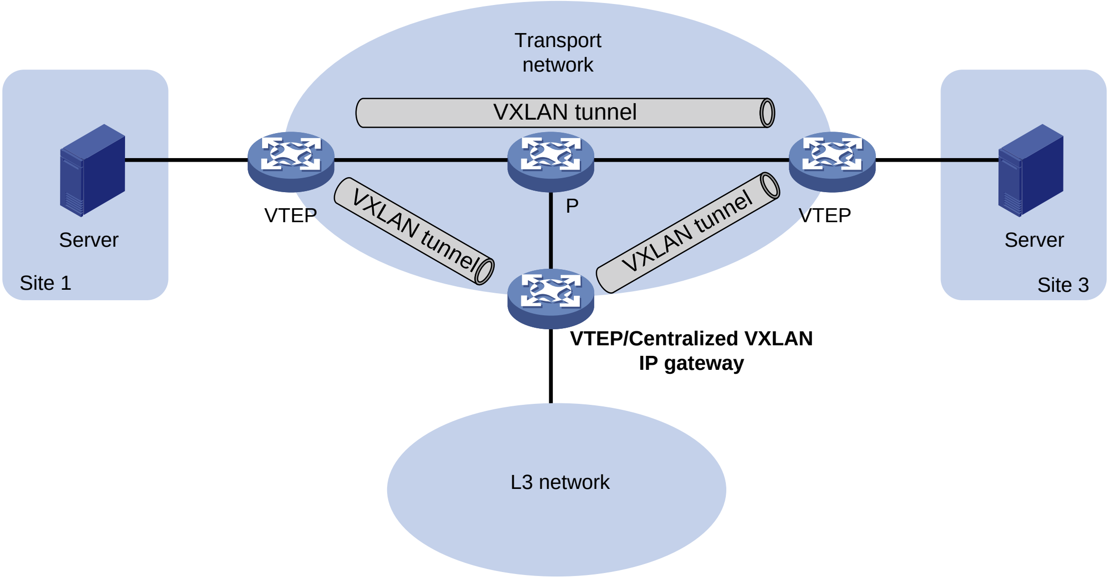
Figure 8 VXLAN IP gateway separated from VTEPs



Centralized VXLAN IP gateway deployment

As shown in Figure 9, a VTEP acts as a gateway for VMs in the VXLANs. The VTEP both terminates the VXLANs and performs Layer 3 forwarding for the VMs.

Figure 9 Centralized VXLAN IP gateway placement design



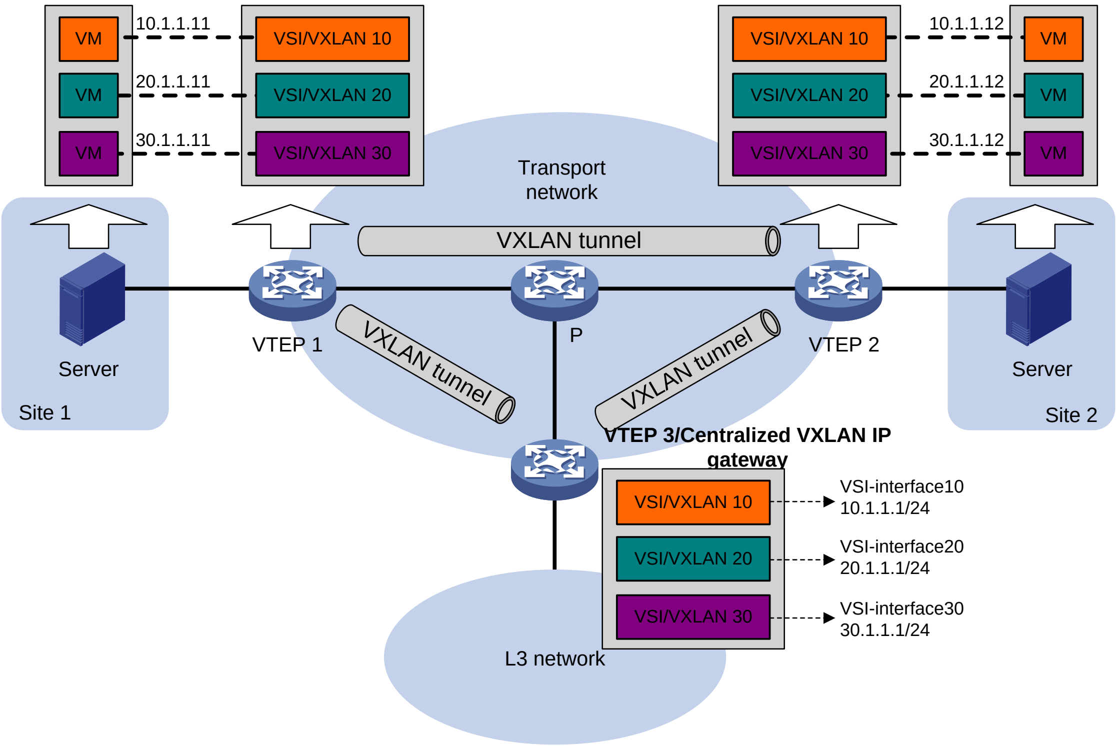
As shown in Figure 10, the network uses the following process to forward Layer 3 traffic from VM 10.1.1.11 to the Layer 3 network:

1. The VM sends an ARP request to obtain the MAC address of the gateway (VTEP 3) at 10.1.1.1.
2. VTEP 1 floods the ARP request to all remote VTEPs.
3. VTEP 3 de-encapsulates the ARP request, creates an ARP entry for the VM, and sends an ARP reply to the VM.

4. VTEP 1 forwards the ARP reply to the VM.
5. The VM learns the MAC address of the gateway, and sends the Layer 3 traffic to the gateway.
6. VTEP 3 removes the VXLAN encapsulation and inner Ethernet header for the traffic, and forwards the traffic to the destination node.

Inter-VXLAN forwarding is the same as this process except for the last step. At the last step of inter-VLAN forwarding, the gateway replaces the source-VXLAN encapsulation with the destination-VXLAN encapsulation, and then forwards the traffic.

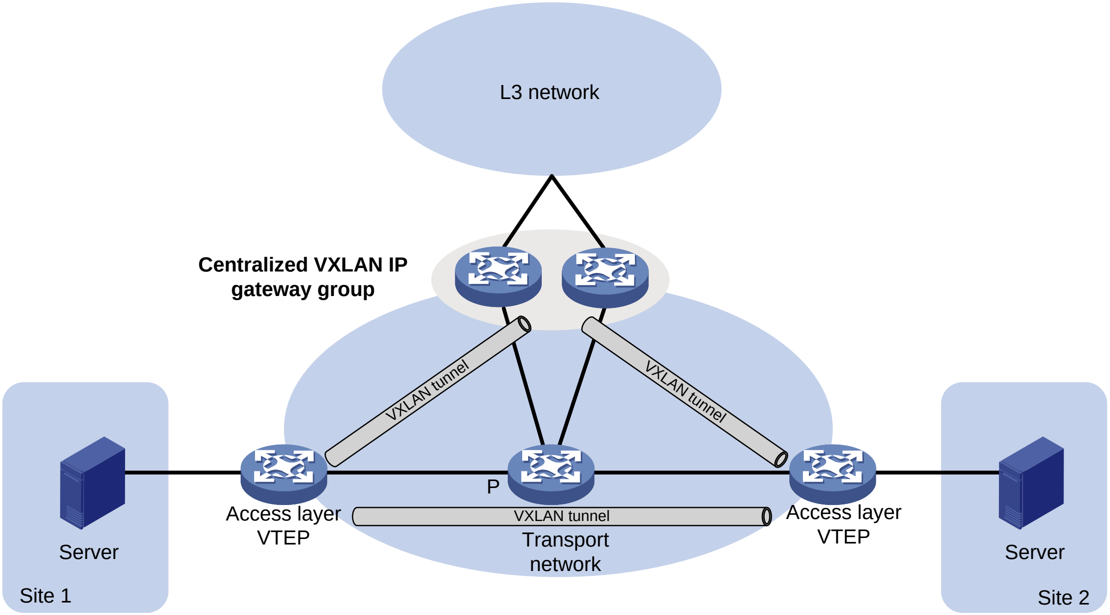
Figure 10 Example of centralized VXLAN IP gateway deployment



Centralized VXLAN gateway group deployment

As shown in Figure 11, a VTEP group uses redundant centralized VXLAN IP gateways to provide reliable gateway services for VMs in the VXLANs. All member VTEPs in the group participate in Layer 3 forwarding and load share traffic between the Layer 3 network and the VXLANs. This design distributes processing among multiple VTEPs and prevents single points of failure.

Figure 11 Example of centralized VXLAN IP gateway group deployment



The VTEP group is a virtual gateway that provides services at a group IP address. Access layer VTEPs set up VXLAN tunnels to the group IP address for data traffic forwarding. Each VTEP in the group automatically uses its member IP address to set up tunnels to the other member VTEPs and access layer VTEPs. The tunnels are used to transmit protocol packets and synchronize ARP entries.

Distributed VXLAN IP gateway deployment

About distributed VXLAN IP gateway deployment

As shown in Figure 12, each site's VTEP acts as a gateway to perform Layer 3 forwarding for the VXLANs of the local site. A VTEP acts as a border gateway to the Layer 3 network for the VXLANs.

Figure 12 Distributed VXLAN IP gateway placement design

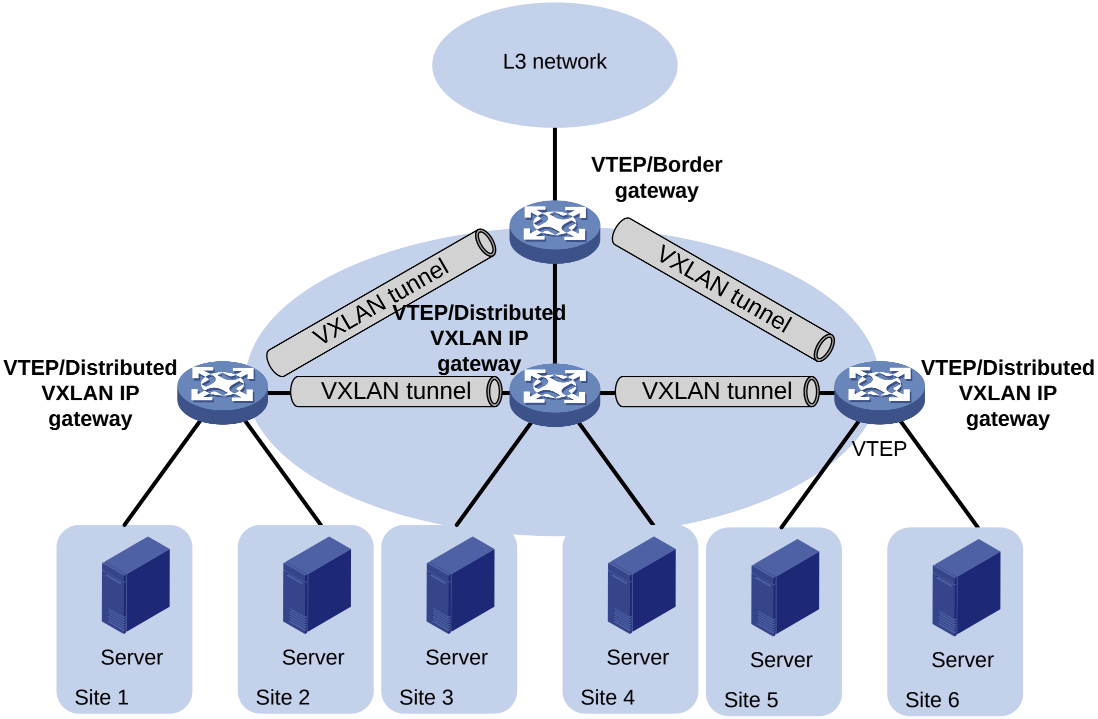
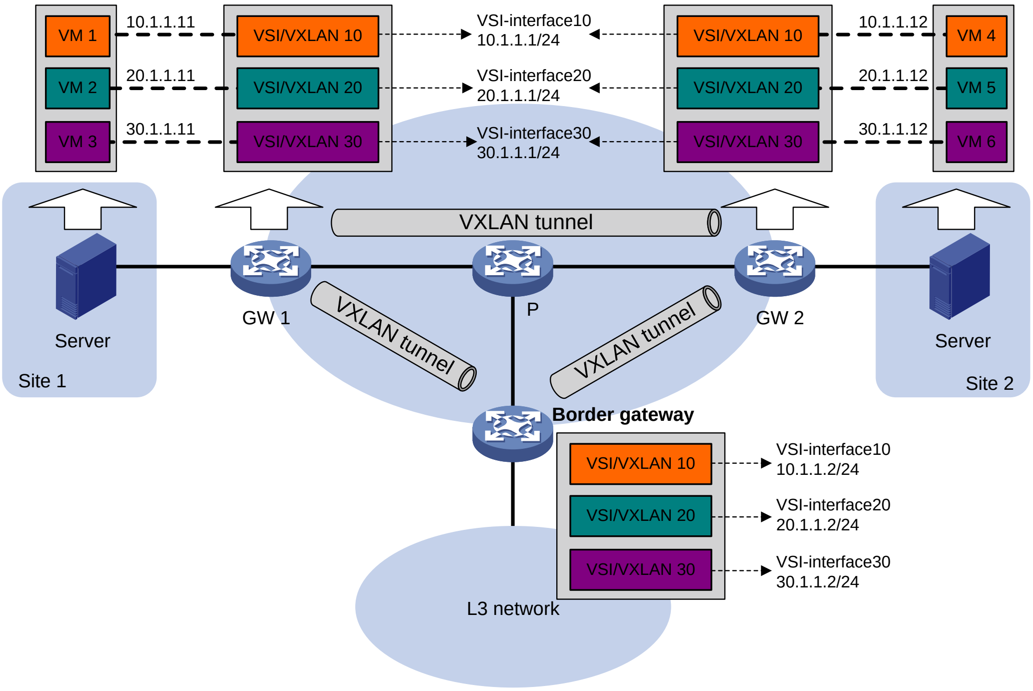


Figure 13 shows an example of distributed VXLAN IP gateway deployment. Create VSI interfaces on each distributed VXLAN IP gateway and the border gateway as gateway interfaces. Assign the same IP address to the same VSI interface on the distributed VXLAN IP gateways. You must enable local proxy ARP or local ND proxy on a distributed VXLAN IP gateway. The gateway performs Layer 3 forwarding based on ARP or ND entries. The following sections use distributed VXLAN IP gateways enabled with the local proxy ARP or local ND proxy feature to describe the forwarding processes for intra-VXLAN traffic, inter-VXLAN traffic, and traffic from a VXLAN to an external network.

A distributed VXLAN IP gateway can generate ARP or ND entries by a variety of methods. The following sections use dynamically learned ARP or ND entries to describe the forwarding processes.

Figure 13 Example of distributed VXLAN IP gateway deployment



Intra-VXLAN traffic forwarding between sites

As shown in Figure 13, the network uses the following process to forward traffic in a VXLAN between sites (for example, from VM 1 to VM 4 in VXLAN 10):

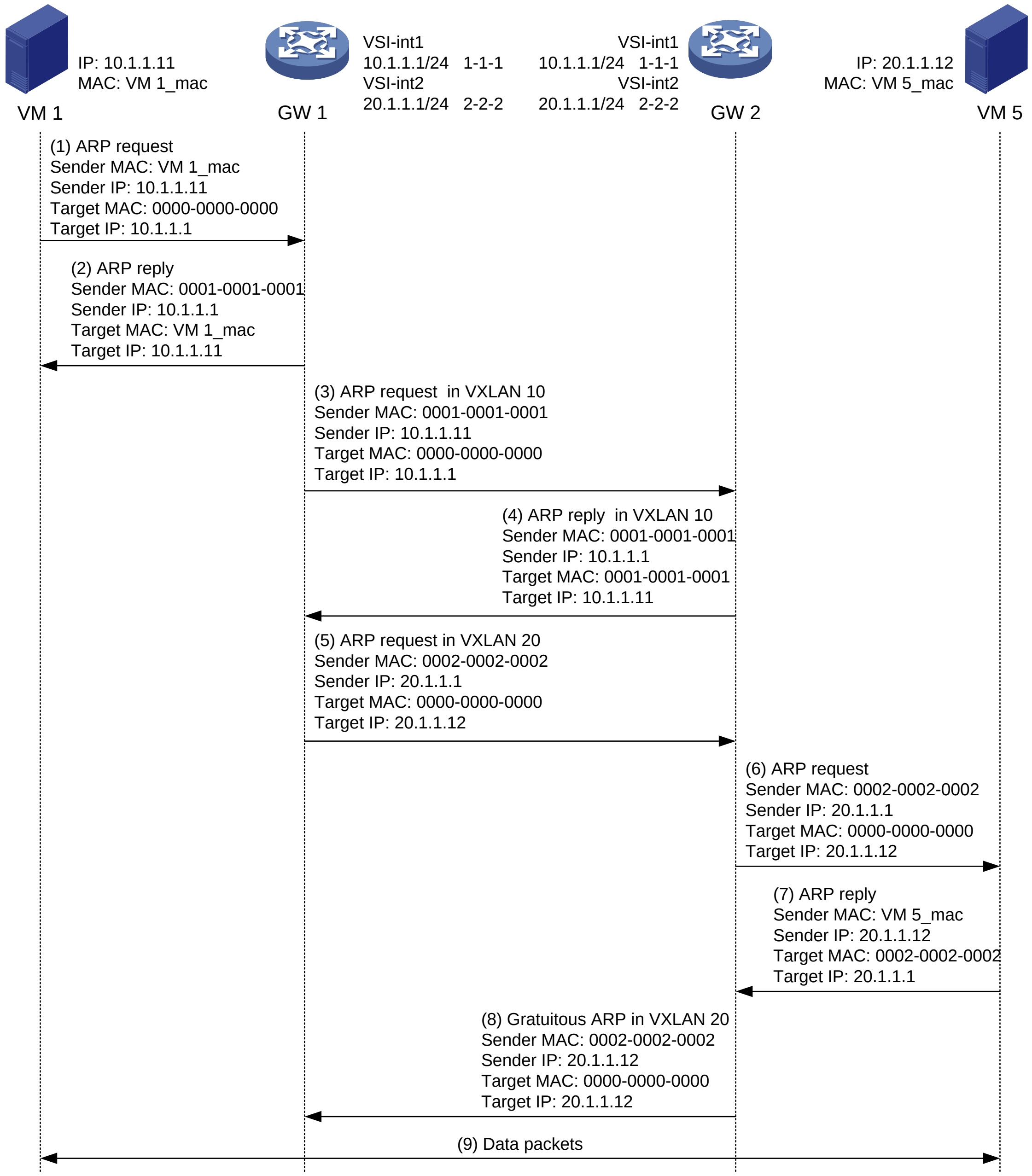
1. VM 1 sends an ARP request to obtain the MAC address of VM 4.
2. GW 1 performs the following operations:
   a. Creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10).
   b. Replaces the sender MAC address of the ARP request with the MAC address of VSI-interface 10, and then floods the request to all sites in VXLAN 10.
3. VM 1 creates an ARP entry for VM 4. The MAC address in the entry is the MAC address of VSI-interface 10 on GW 1.
4. GW 2 (the VTEP for VM 4) performs the following operations:
   a. De-encapsulates the ARP request and creates an ARP entry for VM 1. The entry contains VM 1's IP address (10.1.1.11), the MAC address of VSI-interface 10 on GW 1, and the incoming tunnel interface.
   b. Replaces the sender MAC address of the request with the MAC address of VSI-interface 10 on GW 2, and then floods the request to the local site in VXLAN 10.
5. VM 4 creates an ARP entry for VM 1, and then sends a reply to GW 2. The MAC address in the ARP entry is the MAC address of VSI-interface 10 on GW 2.
6. GW 2 performs the following operations:
   a. Creates an ARP entry for VM 4.
   b. Replaces the sender MAC address of the reply with the MAC address of VSI-interface 10 on GW 2, and sends the reply to GW 1.
7. GW 1 de-encapsulates the ARP reply and creates an ARP entry for VM 4. The entry contains VM 4's IP address (10.1.1.12), the MAC address of VSI-interface 10 on GW 2, and the incoming tunnel interface.
8. For subsequent traffic between VM 1 and VM 4, GW 1 and GW 2 use their respective ARP tables to make the forwarding decision.

Inter-VXLAN traffic forwarding between sites

As shown in Figure 14, the network uses the following process to forward traffic between VXLANs (for example, from VM 1 in VXLAN 10 to VM 5 in VXLAN 20):

1. VM 1 sends an ARP request to obtain the MAC address of the gateway at 10.1.1.1.
2. GW 1 creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10) so VM 1 will send the packets destined for VM 5 to GW 1.
3. GW 1 sends an ARP request to the local and remote sites in VXLAN 10. In the ARP request, the sender IP address is 10.1.1.11, and the sender MAC address is the MAC address of VSI-interface 10 on GW 1.
4. GW 2 performs the following operations:
    a. De-encapsulates the ARP request and creates an ARP entry for VM 1. The entry contains IP address 10.1.1.11 and MAC address of VSI-interface 10 on GW 1, and the incoming tunnel interface.
    b. Replaces the sender MAC address of the request with the MAC address of VSI-interface 10 on GW 2, and then floods the request to the local site in VXLAN 10.
    c. Sends an ARP reply to GW 1. The reply contains IP address 10.1.1.1 and MAC address of VSI-interface 10 on GW 2).
5. When sending an ARP request in VXLAN 10, GW 1 also sends an ARP request to the local and remote sites in VXLAN 20 to obtain the MAC address of VM 5. In the ARP request, the sender IP address is 20.1.1.1, and the sender MAC address is the MAC address of VSI-interface 20 on GW 1.
6. GW 2 de-encapsulates the ARP request of VXLAN 20, replaces the sender MAC address of the request with the MAC address of VSI-interface 20 on GW 2, and then floods the request to the local site in VXLAN 20.
7. VM 5 creates an ARP entry for GW 2, and then sends a reply to GW 2. The entry contains IP address 20.1.1.1 and MAC address of VSI-interface 20 on GW 2.
8. GW 2 performs the following operations:
    a. Creates an ARP entry for VM 5.
    b. Sends a gratuitous ARP packet to the local and remote sites. In the packet, the sender IP address is 20.1.1.12, and the sender MAC address is the MAC address of VSI-interface 20 on GW 2.
9. GW 1 de-encapsulates the gratuitous ARP packet and creates an ARP entry for VM 5. The entry contains VM 5's IP address 20.1.1.12, the MAC address of VSI-interface 20 on GW 2, and the incoming tunnel interface.
10. For subsequent traffic between VM 1 and VM 5, GW 1 and GW 2 use their respective ARP tables to make the forwarding decision.

Figure 14 Inter-VXLAN traffic forwarding between sites



VXLAN-to-external network traffic forwarding

As shown in Figure 13, the network uses the following process to forward traffic from a VXLAN to the Layer 3 network (for example, from VM 1 to the host at 50.1.1.1):

1. VM 1 sends an ARP request to obtain the MAC address of the gateway at 10.1.1.1.
2. GW 1 creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10).
3. VM 1 sends a packet destined for the host to GW 1.
4. GW 1 performs the following operations:
   a. Searches the IP routing policies or routing table for the next hop. In this example, the next hop for the packet is 10.1.1.2 (the border gateway).

    b. Floods an ARP request to the local and remote sites in VXLAN 10 to obtain the MAC address of 10.1.1.2.

5. The border gateway de-encapsulates the ARP request, creates an ARP entry for GW 1, and tunnels a reply to GW 1.
6. GW 1 de-encapsulates the ARP reply and creates an ARP entry for 10.1.1.2.
7. GW 1 sends the packet destined for the host to the border gateway.
8. The border gateway de-encapsulates the packet and forwards it to the host.

Restrictions and guidelines: VXLAN IP gateway configuration

Do not configure both centralized VXLAN IP gateway settings and centralized VXLAN IP gateway group settings on a device.

As a best practice to avoid forwarding failure, set a large MTU on the traffic outgoing interfaces for VXLAN tunnels on VXLAN IP gateways.

VXLAN IP gateway tasks at a glance

To configure a VXLAN IP gateway, perform the following tasks:

1. Configure a VXLAN IP gateway

   Choose one of the following tasks:

   - Configuring a centralized VXLAN IP gateway
   - Configuring a centralized VXLAN IP gateway group
   - Configuring a distributed VXLAN IP gateway
2. (Optional.) Disabling remote ARP learning for VXLANs
3. (Optional.) Configuring a VSI interface

Prerequisites for VXLAN IP gateway configuration

Before you configure a centralized or distributed VXLAN IP gateway, you must perform the following tasks on VTEPs:

- Enable Layer 3 forwarding for VXLANs.
- Create VSIs and VXLANs.

Configuring a centralized VXLAN IP gateway

Restrictions and guidelines

Do not execute the **`local-proxy-arp enable`** command on the VSI interfaces of a centralized VXLAN IP gateway.

Configuring a gateway interface on a centralized VXLAN IP gateway

1. Enter system view.

system-view

2. Create a VSI interface and enter VSI interface view.

   interface vsi-interface *vsi-interface-id*

3. Assign an IPv4 address to the VSI interface.

   ip address *ip-address* { *mask* | *mask-length* }

   By default, no IPv4 address is assigned to a VSI interface.

4. Return to system view.

   quit

5. Enter VSI view.

   vsi *vsi-name*

6. Specify a gateway interface for the VSI.

   gateway vsi-interface *vsi-interface-id*

   By default, no gateway interface is specified for a VSI.

Assigning a subnet to a VSI

About this task

Perform this task on VSIs that share a gateway interface. This task enables the VSI interface to identify the VSI of a packet.

You can assign a maximum of eight IPv4 subnets to a VSI. Make sure these subnets are on the same network as one of the IP addresses on the gateway interface.

For VSIs that share a gateway interface, the subnets must be unique.

If you remove the gateway interface from the VSI, the VSI's subnet settings are automatically deleted.

Procedure

1. Enter system view.

   system-view

2. Enter VSI view.

   vsi *vsi-name*

3. Assign a subnet to the VSI.

   gateway subnet *ipv4-address wildcard-mask*

   By default, no subnet exists on a VSI.

Configuring a centralized VXLAN IP gateway group

Configuring a VTEP group

Restrictions and guidelines

Make sure the member VTEPs use the same VXLAN settings.

Procedure

1. Enter system view.

   system-view

2. Create a VSI interface and enter VSI interface view.

   interface vsi-interface *vsi-interface-id*

   This interface will be used as the gateway interface for the VSI.

3. Assign an IP address to the VSI interface.

   ip address *ip-address* { *mask* | *mask-length* }

   By default, no IP address is assigned to a VSI interface.

   You must assign the same IP address to the VSI interface on each VTEP in the VTEP group.

4. Assign a MAC address to the VSI interface.

   mac-address *mac-address*

   By default, a VSI interface does not have a MAC address.

   You must assign the same MAC address to the VSI interface on each VTEP in the VTEP group.

5. Return to system view.

   quit

6. Enter VSI view.

   vsi *vsi-name*

7. Specify the VSI interface as the gateway interface for the VSI.

   gateway vsi-interface *vsi-interface-id*

   By default, no gateway interface is specified for a VSI.

8. Return to system view.

   quit

9. Assign the local VTEP to a VTEP group and specify a member IP address for the VTEP.

   vtep group *group-ip* **member local** *member-ip*

   By default, a VTEP is not assigned to any VTEP group.

   The specified member IP address must already exist on the local VTEP and be unique in the VTEP group. You must configure a routing protocol to advertise the IP address to the transport network.

10. Specify the member IP address of all the other VTEPs in the VTEP group.

    vtep group *group-ip* **member remote** *member-ip*&<1-8>

    By default, the list of remote VTEPs is not configured.

Specifying a VTEP group as the gateway for an access layer VTEP

Prerequisites

Before you specify a VTEP group on an access layer VTEP, perform the following tasks on the VTEP:

- Enable Layer 2 forwarding for VXLANs.
- Configure VSIs and VXLANs.
- Set up VXLAN tunnels to remote sites and the VTEP group, and assign the tunnels to VXLANs.

Procedure

1. Enter system view.

   system-view

2. Specify a VTEP group and all its member VTEPs.

   vtep group *group-ip* **member remote** *member-ip*&<1-8>

By default, no VTEP group is specified.

Perform this task to specify all member VTEPs in the VTEP group.

Configuring a distributed VXLAN IP gateway

Restrictions and guidelines for distributed VXLAN IP gateway configuration

For a VXLAN that requires access to the external network, specify the VXLAN's VSI interface on the border gateway as the next hop by using one of the following methods:

- Configure a static route.
- Configure a routing policy, and apply the policy by using the **`apply default-next-hop`** command. For more information about configuring routing policies, see routing policy configuration in *Layer 3—IP Routing Configuration Guide*.

Make sure a VSI interface uses the same MAC address to provide service on distributed VXLAN IP gateways connected to IPv4 sites. Make sure a VSI interface uses different link-local addresses to provide service on distributed VXLAN IP gateways connected to both IPv4 and IPv6 sites.

Configuring a gateway interface on a distributed VXLAN IP gateway

1. Enter system view.

   `system-view`

2. Create a VSI interface and enter VSI interface view.

   `interface vsi-interface` *`vsi-interface-id`*

3. Assign an IP address to the VSI interface.

   IPv4:

   `ip address` *`ip-address`* `{` *`mask`* `|` *`mask-length`* `} [` **`sub`** `]`

   IPv6:

   See IPv6 basics in *Layer 3—IP Services Configuration Guide*.

   By default, no IP address is assigned to a VSI interface.

4. Specify the VSI interface as a distributed gateway.

   `distributed-gateway local`

   By default, a VSI interface is not a distributed gateway.

5. Enable local proxy ARP or local ND proxy.

   IPv4:

   `local-proxy-arp enable` `[` **`ip-range`** *`startIP`* **`to`** *`endIP`* `]`

   By default, local proxy ARP is disabled.

   For more information about this command, see proxy ARP commands in *Layer 3—IP Services Command Reference*.

   IPv6:

   `local-proxy-nd enable`

   By default, local ND proxy is disabled.

   For more information about this command, see IPv6 basics commands in *Layer 3—IP Services Command Reference*.

6. Bring up the VSI interface.

   undo shutdown

   By default, a VSI interface is up.

7. Return to system view.

   quit

8. Enter VSI view.

   vsi *vsi-name*

9. Specify the VSI interface as the gateway interface for the VSI.

   gateway vsi-interface *vsi-interface-id*

   By default, no gateway interface is specified for a VSI.

Enabling dynamic ARP entry synchronization for distributed VXLAN IP gateways

About this task

When local proxy ARP is enabled on distributed VXLAN IP gateways, enable this feature for all gateways to have the same ARP entries.

A controller can also synchronize ARP entries among distributed VXLAN IP gateways. When you use a controller, do not enable dynamic ARP entry synchronization.

Procedure

1. Enter system view.

   system-view

2. Enable dynamic ARP entry synchronization for distributed VXLAN IP gateways.

   arp distributed-gateway dynamic-entry synchronize

   By default, dynamic ARP entry synchronization is disabled for distributed VXLAN IP gateways.

Assigning a subnet to a VSI

About this task

Perform this task on VSIs that share a gateway interface. This task enables the VSI interface to identify the VSI of a packet.

You can assign a maximum of eight IPv4 and IPv6 subnets to a VSI. Make sure these subnets are on the same network as one of the IP addresses on the gateway interface.

For VSIs that share a gateway interface, the subnets must be unique.

If you remove the gateway interface from the VSI, the VSI's subnet settings are automatically deleted.

Procedure

1. Enter system view.

   system-view

2. Enter VSI view.

   vsi *vsi-name*

3. Assign a subnet to the VSI.

   gateway subnet { *ipv4-address wildcard-mask* | *ipv6-address prefix-length* }

By default, no subnet exists on a VSI.

Disabling remote ARP learning for VXLANs

About this task

By default, the device learns ARP information of remote user terminals from packets received on VXLAN tunnel interfaces. To save resources on VTEPs in an SDN transport network, you can temporarily disable remote ARP learning when the controller and VTEPs are synchronizing entries. After the entry synchronization is completed, enable remote ARP learning.

Restrictions and guidelines

As a best practice, disable remote ARP learning for VXLANs only when the controller and VTEPs are synchronizing entries.

Procedure

1. Enter system view.

   `system-view`

2. Disable remote ARP learning for VXLANs.

   `vxlan tunnel arp-learning disable`

   By default, remote ARP learning is enabled for VXLANs.

Configuring a VSI interface

Configuring optional parameters for a VSI interface

1. Enter system view.

   `system-view`

2. Enter VSI interface view.

   `interface vsi-interface` *`vsi-interface-id`*

3. Assign a MAC address to the VSI interface.

   `mac-address` *`mac-address`*

   By default, the MAC address of a VSI interface is the bridge MAC address.

4. Configure the description of the VSI interface.

   `description` *`text`*

   The default description of a VSI interface is *`interface-name`* plus **`Interface`** (for example, **`Vsi-interface100 Interface`**).

5. Set the MTU for the VSI interface.

   `mtu` *`mtu-value`*

   The default MTU of a VSI interface is 1500 bytes.

6. Set the expected bandwidth for the VSI interface.

   `bandwidth` *`bandwidth-value`*

   The default expected bandwidth (in kbps) equals the interface baudrate divided by 1000.

   The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

Restoring the default settings of the VSI interface

Restrictions and guidelines

△ **CAUTION:**

This operation might interrupt ongoing network services. Make sure you are fully aware of the impact of this operation when you perform it on a live network.

This operation might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **`display this`** command in interface view to identify these commands. Use their **`undo`** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

Procedure

1. Enter system view.
   `system-view`
2. Enter VSI interface view.
   `interface vsi-interface` *`vsi-interface-id`*
3. Restore the default settings of the VSI interface.
   `default`

Display and maintenance commands for VXLAN IP gateways

Execute **`display`** commands in any view and **`reset`** commands in user view.

Task	Command
Display information about VSI interfaces.	**`display interface`** [**`vsi-interface`** [*`vsi-interface-id`*]] [**`brief`** [**`description`** | **`down`**]]
Clear statistics on VSI interfaces.	**`reset counters interface`** [**`vsi-interface`** [*`vsi-interface-id`*]]

Configuring the VTEP as an OVSDB VTEP

About OVSDB VTEP

An NSFOCUS network virtualization controller can use the Open vSwitch Database (OVSDB) management protocol to deploy and manage VXLANs on VTEPs. To work with a controller, you must configure the VTEP as an OVSDB VTEP.

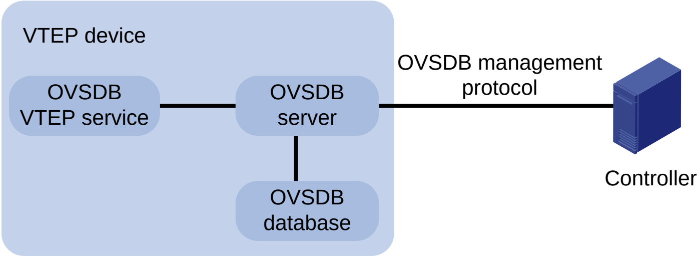
Working mechanisms

As shown in Figure 15, an OVSDB VTEP stores all of its VXLAN settings in the form of entries in an OVSDB database. The OVSDB database, OVSDB VTEP service, and the controller interact through the OVSDB server. The controller communicates with the OVSDB server through the OVSDB protocol to manage the OVSDB database. The OVSDB VTEP service reads and writes data in the OVSDB database through the OVSDB server.

The OVSDB VTEP service performs the following operations to manage the VXLAN settings on the VTEP:

- Converts data in the OVSDB database into VXLAN configuration and deploys the configuration to the VTEP. For example, create or remove a VXLAN or VXLAN tunnel.
- Adds site-facing interface information and the global source address of VXLAN tunnels to the OVSDB database. The information is reported to the controller by the OVSDB server.

Figure 15 OVSDB network model



Protocols and standards

RFC 7047, *The Open vSwitch Database Management Protocol*

Restrictions: Hardware compatibility with OVSDB VTEP

Models	**OVSDB VTEP compatibility**
NFNX5-HD6480, NFNX3-HDB3280, NFNX5-T6280, NFNX5-HD5280, NFNX3-HDB1780, NFNX3-HDB3080, NFNX3-HDB1180, NFNX3-HDB1480	Yes
NFNX3-HDB680, NFNX3-HDB1080	No

Restrictions and guidelines: OVSDB VTEP configuration

You can configure a VTEP both at the CLI and through a controller. As a best practice, do not manually remove the VXLAN configuration issued by the controller.

OVSDB VTEP tasks at a glance

To configure OVSDB VTEPs, perform the following tasks:

1. Setting up an OVSDB connection to a controller
   - Configuring active SSL connection settings
   - Configuring passive SSL connection settings
   - Configuring active TCP connection settings
   - Configuring passive TCP connection settings
2. Enabling the OVSDB server
3. Enabling the OVSDB VTEP service
4. Specifying a global source address for VXLAN tunnels
5. Specifying a VTEP access port
6. Enabling flood proxy on multicast VXLAN tunnels

   If you use a flood proxy server, you must enable flood proxy globally on multicast tunnels.

Prerequisites for OVSDB VTEP configuration

Before you configure the VTEP as an OVSDB VTEP, enable L2VPN by using the **`l2vpn enable`** command.

Before you set up SSL connections to controllers, you must configure SSL as described in *Security Configuration Guide*.

Setting up an OVSDB connection to a controller

About OVSDB connection types

The OVSDB server supports the following types of OVSDB connections:

- **Active SSL connection**—The OVSDB server initiates an SSL connection to the controller.
- **Passive SSL connection**—The OVSDB server accepts the SSL connection from the controller.
- **Active TCP connection**—The OVSDB server initiates a TCP connection to the controller.
- **Passive TCP connection**—The OVSDB server accepts the TCP connection from the controller.

Restrictions and guidelines for OVSDB controller connection setup

When you set up OVSDB connections, follow these restrictions and guidelines:

- You can set up multiple OVSDB connections. For the device to establish the connections, you must enable the OVSDB server. You must disable and then re-enable the OVSDB server if it has been enabled.
- You must specify the same PKI domain and CA certificate file for all active and passive SSL connections.

Prerequisites for OVSDB controller connection setup

Make sure you have configured a PKI domain before specifying it for SSL. For more information about configuring a PKI domain, see *Security Configuration Guide*.

Configuring active SSL connection settings

1. Enter system view.

   system-view

2. Specify a PKI domain for SSL.

   ovsdb server pki domain *domain-name*

   By default, no PKI domain is specified for SSL.

3. (Optional.) Specify a CA certificate file for SSL.

   ovsdb server bootstrap ca-certificate *ca-filename*

   By default, SSL uses the CA certificate file in the PKI domain.

   If the specified CA certificate file does not exist, the device obtains a self-signed certificate from the controller. The obtained file uses the name specified for the *ca-filename* argument.

4. Set up an active SSL connection.

   ovsdb server ssl ip *ip-address* **port** *port-number*

   By default, the device does not have active OVSDB SSL connections.

   You can set up a maximum of eight OVSDB SSL connections.

Configuring passive SSL connection settings

1. Enter system view.

   system-view

2. Specify a PKI domain for SSL.

   ovsdb server pki domain *domain-name*

   By default, no PKI domain is specified for SSL.

3. (Optional.) Specify a CA certificate file for SSL.

   ovsdb server bootstrap ca-certificate *ca-filename*

   By default, SSL uses the CA certificate file in the PKI domain.

   If the specified CA certificate file does not exist, the device obtains a self-signed certificate from the controller. The obtained file uses the name specified for the *ca-filename* argument.

4. Enable the device to listen for SSL connection requests.

   ovsdb server pssl [**port** *port-number*]

   By default, the device does not listen for SSL connection requests.

   You can specify only one port to listen for OVSDB SSL connection requests.

Configuring active TCP connection settings

1. Enter system view.

   system-view

2. Set up an active TCP connection.

   ovsdb server tcp ip *ip-address* **port** *port-number*

   By default, the device does not have active OVSDB TCP connections.

   You can set up a maximum of eight active OVSDB TCP connections.

Configuring passive TCP connection settings

1. Enter system view.

   system-view

2. Enable the device to listen for TCP connection requests.

   ovsdb server ptcp [**port** *port-number*]

   By default, the device does not listen for TCP connection requests.

   You can specify only one port to listen for OVSDB TCP connection requests.

Enabling the OVSDB server

Prerequisites

Make sure you have complete OVSDB connection setup before you enable the OVSDB server. If you change OVSDB connection settings after the OVSDB server is enabled, you must disable and then re-enable the OVSDB server for the change to take effect.

Procedure

1. Enter system view.

   system-view

2. Enable the OVSDB server.

   ovsdb server enable

   By default, the OVSDB server is disabled.

Enabling the OVSDB VTEP service

1. Enter system view.

   system-view

2. Enable the OVSDB VTEP service.

   vtep enable

   By default, the OVSDB VTEP service is disabled.

Specifying a global source address for VXLAN tunnels

About this task

The VTEP reports the global VXLAN tunnel source address to the controller for VXLAN tunnel setup.

Restrictions and guidelines

For correct VXLAN deployment and VTEP management, do not manually specify tunnel-specific source addresses for VXLAN tunnels if OVSDB is used.

Procedure

1. Enter system view.

   `system-view`

2. Specify a global source address for VXLAN tunnels.

   `tunnel global source-address` { *`ipv4-address`* | **`ipv6`** *`ipv6-address`* }

   By default, no global source address is specified for VXLAN tunnels.

Specifying a VTEP access port

About this task

For the controller to manage a site-facing interface, you must specify the interface as a VTEP access port.

Procedure

1. Enter system view.

   `system-view`

2. Enter interface view.

   `interface` *`interface-type interface-number`*

3. Specify the interface as a VTEP access port.

   `vtep access port`

   By default, an interface is not a VTEP access port.

Enabling flood proxy on multicast VXLAN tunnels

About this task

If you use a flood proxy server, you must enable flood proxy globally on multicast tunnels. Then the multicast tunnels are converted into flood proxy tunnels. The VTEP sends broadcast, multicast, and unknown unicast traffic for a VXLAN to the flood proxy server through the tunnels. The flood proxy server then replicates and forwards flood traffic to remote VTEPs.

Procedure

1. Enter system view.

   `system-view`

2. Enable flood proxy on multicast VXLAN tunnels.

   `vxlan tunnel flooding-proxy`

   By default, flood proxy is disabled on multicast VXLAN tunnels.

NSFOCUS Firewall Series

NF Service Chain Instance Configuration Guide

NSFOCUS Technologies, Inc.

Document version: 6W600-20230221

Preface

- This configuration guide describes the fundamentals and configuration procedures for Service Chain instance features.
- Security zones, security policies, object groups, object policies, APR, and session management.
- User access and authentication features (such as AAA, portal, user identification, password control, and PKI).
- Data security features (such as public key management, SSL, SSH, and crypto engine).
- Attack protection features (such as ASPF, connection limit, IP source guard, ARP attack protection, ND attack defense, uRPF, attack detection and prevention, and IP-MAC binding).

This preface includes the following topics about the documentation:

- Audience.
- Conventions.

Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

Conventions

The following information describes the conventions used in the documentation.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.
[x | y | ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

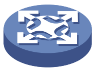
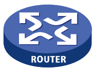
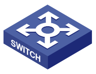
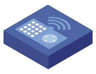
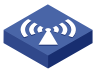
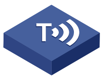
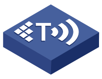
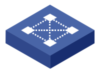
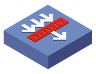
GUI conventions

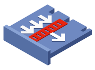
Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the **New User** window opens; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ **WARNING!**	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ **CAUTION:**	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ **IMPORTANT:**	An alert that calls attention to essential information.
NOTE:	An alert that contains additional or supplementary information.
💡 **TIP:**	An alert that provides helpful information.

Network topology icons

Convention	Description
	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.

Convention	Description
	Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.

Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

Contents

Configuring service chains

About service chains

The service chain technology is used to guide network service packets through service nodes. The technology is used in combination with software defined network (SDN) for service orchestration. Service packets in a service chain are forwarded and processed by service nodes in a specific order.

Inter-device service chain

Inter-device service chains apply only to VXLAN packets. Service chain policies are used to control packets in the Overlay network as follows:

1. Encapsulate IP packets into VXLAN packets for service chain processing.
2. Deliver the VXLAN packets through all service nodes in a service chain.

You can deploy service chain policies from a VCF controller to tenants by context through OpenFlow.

For more information about VXLAN, see *VXLAN Configuration Guide*. For more information about VCF controllers, see the VCF controller manuals.

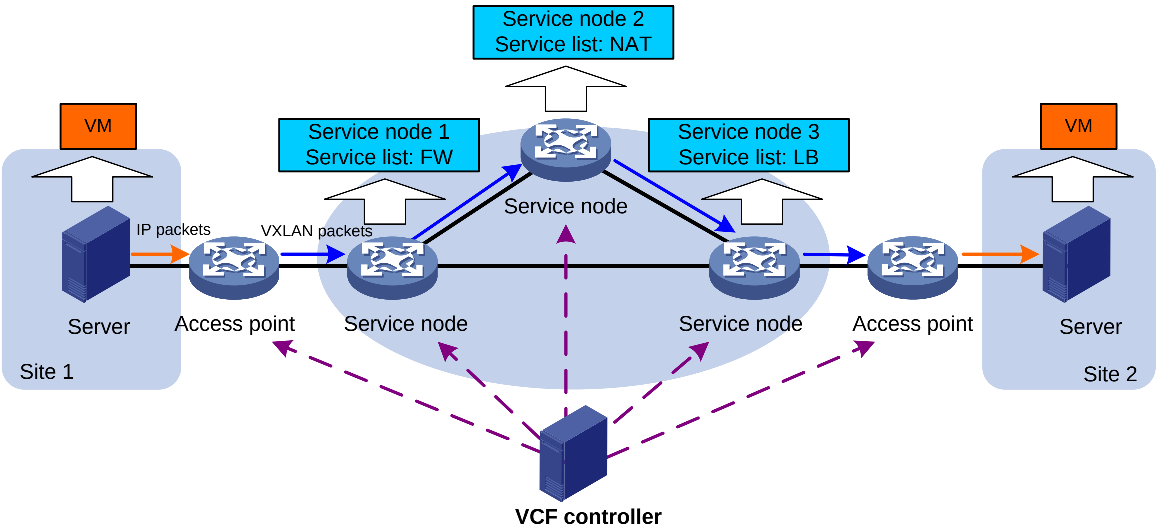
NOTE:

A service chain applied to contexts of different users on a device is an inter-device service chain.

Network model

As shown in Figure 1, an inter-device service chain includes access points and service nodes.

Figure 1 Inter-device service chain network diagram



Access point

An access point is a VXLAN tunnel end point (VTEP) that processes packets based on the service chain policy deployed by a VCF controller.

Service node

A service node is a physical device or a network function virtualization (NFV) device that applies services in a service list to the received VXLAN packets. An inter-device service chain can have multiple service nodes.

A service list on a service node specifies the types and order of services to be applied to the VXLAN packets. Available services include FW and LB.

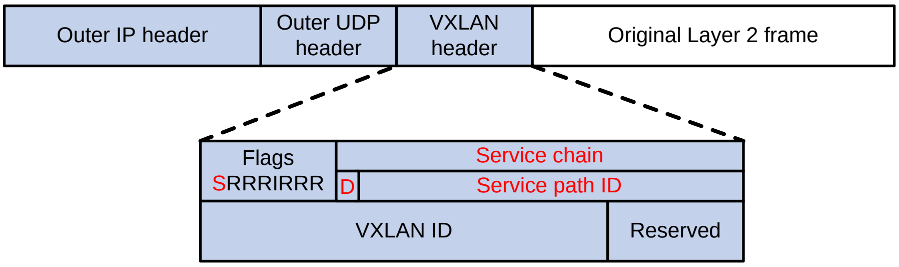
An inter-device service chain supports the following special types of service nodes:

- **Head node**—The first service node in a service chain that processes the VXLAN packets.
- **End node**—The last service node in a service chain that processes the VXLAN packets.

Packet format

Inter-device service chain packets use the VXLAN header. Figure 2 shows the format of a service chain packet.

Figure 2 Service chain packet format



The service chain modifies the following fields in a VXLAN header:

- **Flags**—If the **S** bit is 1, the **Service chain** field is valid. If the **S** bit is 0, the **Service chain** field is invalid.
- **Service chain**—A 24-bit field that includes the **D** bit and service path ID. If the **D** bit is 0, the packet is a forward packet. If the **D** bit is 1, the packet is a backward packet. The 23-bit service path ID is used to identify an inter-device service chain.

Operating mechanisms

An inter-device service chain processes a packet in the following procedure:

1. Based on the service chain policy deployed from the VCF controller, the access point in the inbound packet direction (inbound access point) processes the received packet as follows:
   a. Encapsulates the packet into a VXLAN packet.
   b. Forwards the VXLAN packet to the head node.
2. The head node compares the service path ID in the VXLAN header of the received packet with the path IDs of the service chains.
   - If the packet matches a service chain, the head node forwards the packet into the service chain.
   - If the packet does not match any service chain, the head node performs a VXLAN forwarding for the packet.
3. After a service node completes processing the received VXLAN packet, it forwards the packet in one of the following ways:
   - If the IP address of the next service node is specified, the node forwards the packet to the next service node.

- If no next service node address is specified, the current service node is the end node. The service node removes the service path ID field from the packet and forwards the packet to the access point in the outbound packet direction (outbound access point).

4. The outbound access point decapsulates the VXLAN packet and performs an IP forwarding for the packet.

Configuring an inter-device service chain

You can configure an inter-device service chain through VCF controller deployment or at the CLI. This document describes inter-device service chain configuration at the CLI. As a best practice, use a VCF controller to deploy the service chain configuration through a NETCONF session. For more information about the VCF controller deployment method, see the VCF controller configuration guide.

Configuring an access point

You can only use a VCF controller to deploy the service chain configuration to a device that acts as an access point. For more information, see the relevant VCF controller manuals.

Configuring a service node

Restrictions and guidelines

If the device is the end node in an inter-device service chain, you only need to specify the previous service node address.

If the device is the head node in an inter-device service chain, you only need to specify the next service node address.

Procedure

1. Enter system view.

   system-view

2. Create a service chain and enter its view.

   service-chain path *path-id*

3. Specify the next service node address for forward packets.

   next-service-node *ip-address*

   By default, no next service node address is specified for forward packets.

4. Specify the previous service node address for backward packets.

   previous-service-node *ip-address*

   By default, no previous service node address is specified for backward packets.

5. Create a service node and enter its view.

   service function *function-id*

6. Configure a service list.

   service list { **fw** | **lb** } *

Display and maintenance commands for service chains

Execute **display** commands in any view.

Task	Command
Display service chain information.	**display service-chain path** { *path-id* | **all** }
Display service chain statistics.	**display service-chain statistics**

Service chain configuration examples

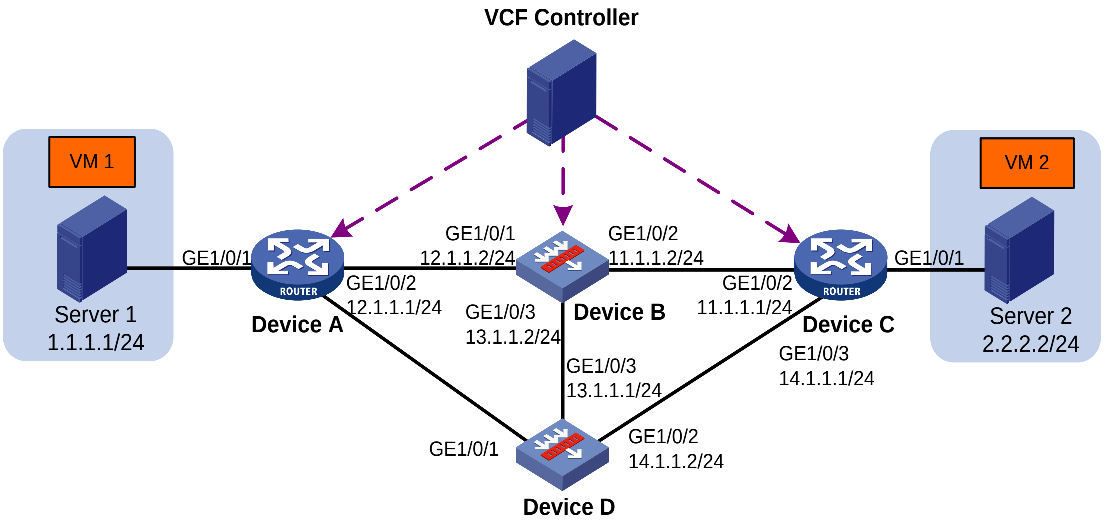
Example: Configuring an inter-device service chain

Network configuration

As shown in Figure 3, Device A and Device C act as access points. Device B and Device D act as service nodes. VM 1 and VM 2 are internal network devices. The inter-device service chain processes the traffic from VM 1 to VM 2 as follows:

1. On Device B, apply the FW service to the traffic.
2. On Device D, apply the LB service to the traffic.

Figure 3 Network diagram



Procedure

1. Configure IP addresses for interfaces to ensure IP connectivity between neighboring nodes. (Details not shown.)
2. Use a VCF controller to deploy service chain policies to Device A and Device C and label the VXLAN packets from VM 1 to VM 2 with service path ID 1. (Details not shown.)
3. Configure Device B:
   a. Assign IP addresses to interfaces:

      # Assign an IP address to interface GigabitEthernet 1/0/1.

```
<DeviceB> system-view
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 12.1.1.2 24
[DeviceB-GigabitEthernet1/0/1] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceB] security-zone name trust
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/1
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceB-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceB-security-zone-Trust] quit
```

c. Configure settings for routing.

This example configures a static route, and the next hop in the route is 12.1.1.1.

```
[DeviceB] ip route-static 1.1.1.0 24 12.1.1.1
```

d. Configure a security policy:

# Configure a rule named **trust-trust** to permit the packets from VM 1 to VM 2.

```
[DeviceB] security-policy ip
[DeviceB-security-policy-ip] rule name trust-trust
[DeviceB-security-policy-ip-1-trust-trust] source-zone trust
[DeviceB-security-policy-ip-1-trust-trust] destination-zone trust
[DeviceB-security-policy-ip-1-trust-trust] source-ip-subnet 1.1.1.0 24
[DeviceB-security-policy-ip-1-trust-trust] destination-ip-subnet 2.2.2.0 24
[DeviceB-security-policy-ip-1-trust-untrust] action pass
[DeviceB-security-policy-ip-1-trust-trust] quit
[DeviceB-security-policy-ip] quit
```

e. Configure service chain settings:

# Create service chain 1 and enter its view.

```
[DeviceB] service-chain path 1
```

# Specify the next service node address as 20.1.1.1, which is the loopback interface address of Device D. The loopback interface is specified as the VXLAN tunnel source interface.

```
[DeviceB-spath1] next-service-node 20.1.1.1
```

# Create service node 1.

```
[DeviceB-spath1] service function 1
```

# Create a service list that includes the FW service.

```
[DeviceB-spath1-func1] service list fw
```

4. Configure Device D:

a. Assign IP addresses to interfaces:

# Assign an IP address to interface GigabitEthernet 1/0/2.

```
<DeviceD> system-view
[DeviceD] interface gigabitethernet 1/0/2
[DeviceD-GigabitEthernet1/0/2] ip address 14.1.1.2 24
[DeviceD-GigabitEthernet1/0/2] quit
```

# Assign IP addresses to other interfaces in the same way. (Details not shown.)

b. Add interfaces to security zones.

```
[DeviceD] security-zone name trust
[DeviceD-security-zone-Trust] import interface gigabitethernet 1/0/2
[DeviceD-security-zone-Trust] import interface gigabitethernet 1/0/3
[DeviceD-security-zone-Trust] quit
```

c. Configure settings for routing.

This example configures a static route, and the next hop in the route is 14.1.1.1.

```
[DeviceD] ip route-static 2.2.2.0 24 14.1.1.1
```

d. Configure a security policy:

# Configure a rule named **trust-trust** to permit the packets from VM 1 to VM 2.

```
[DeviceD] security-policy ip
[DeviceD-security-policy-ip] rule name trust-trust
[DeviceD-security-policy-ip-1-trust-trust] source-zone trust
[DeviceD-security-policy-ip-1-trust-trust] destination-zone trust
[DeviceD-security-policy-ip-1-trust-trust] source-ip-subnet 1.1.1.0 24
[DeviceD-security-policy-ip-1-trust-trust] destination-ip-subnet 2.2.2.0 24
[DeviceD-security-policy-ip-1-trust-trust] action pass
[DeviceD-security-policy-ip-1-trust-untrust] quit
[DeviceD-security-policy-ip] quit
```

e. Configure service chain settings:

# Create service chain 1 and enter its view..

```
[DeviceD] service-chain path 1
```

# Specify the previous service node address as 20.1.1.2, which is the loopback interface address of Device B. The loopback interface is specified as the VXLAN tunnel source interface.

```
[DeviceD-spath1] previous-service-node 20.1.1.2
```

# Create service node 1.

```
[DeviceD-spath1] service function 1
```

# Create a service list that includes the LB service.

```
[DeviceD-spath1-func1] service list lb
```

Verifying the configuration

# Verify the following information: (Details not shown.)

- VXLAN packets from VM 1 to VM 2 are labeled with the path ID of the service chain.
- The FW and LB services are applied to the packets in sequential order.